Circuits, Systems, and Signal Processing (2019) 38:5018–5067
https://doi.org/10.1007/s00034-019-01100-6

Recognition of Spoken Languages from Acoustic Speech Signals Using Fourier Parameters

N. S. Sai Srinivas[1] · N. Sugan[1] · Niladri Kar[1] · L. S. Kumar[1] · Malaya Kumar Nath[1] · Aniruddha Kanhe[1]

Received: 31 December 2017 / Revised: 22 March 2019 / Accepted: 25 March 2019 /
Published online: 4 April 2019

Abstract

Spoken language identification (LID) or spoken language recognition (LR) is defined as the process of recognizing the language from speech utterance. In this paper, a new Fourier parameter (FP) model is proposed for the task of speaker-independent spoken language recognition. The performance of the proposed FP features is analyzed and compared with the legacy mel-frequency cepstral coefficient (MFCC) features. Two multilingual databases, namely Indian Institute of Technology Kharagpur Multilingual Indian Language Speech Corpus (IITKGP-MLILSC) and Oriental Language Recognition Speech Corpus (AP18-OLR), are used to extract FP and MFCC features. Spoken LID/LR models are developed with the extracted FP and MFCC features using three classifiers, namely support vector machines, feed-forward artificial neural networks, and deep neural networks. Experimental results show that the proposed FP features can effectively recognize different languages from speech signals. It can also be observed that the recognition performance is significantly improved when compared to MFCC features. Further, the recognition performance is enhanced when MFCC and FP features are combined.

Keywords AP18-OLR database · AP16-OL7 database · AP17-OL3 database · Artificial neural networks (ANN) · Deep neural networks (DNN) · Fourier parameters (FP) · IITKGP-MLILSC database · Indian languages · Language identification (LID) · Language recognition (LR) · Long short-term memory networks (LSTM) · Mel-frequency cepstral coefficients (MFCC) · Oriental languages · Recurrent neural networks (RNN) · ReliefF feature selection · Speech signal processing · Supervised learning and classification · Support vector machines (SVM)

This work has been supported under 'Special Manpower Development Program for Chip to System Design (SMDP-C2SD) Project,' funded by the Ministry of Electronics and Information Technology (MeitY), Government of India (GOI), under vide Sanction Order and Grant No. 9(1)/2014-MDD(Vol. III).

Extended author information available on the last page of the article

1 Introduction

Spoken language identification (LID) or spoken language recognition (LR) is defined as the process of identifying or recognizing the language from a speech utterance [20]. So far, human beings are considered to be the most highly accurate language recognition systems.[1] It is considered as a trendy research problem for many years and is attracting more attention from the past two decades [18]. It is far different from the traditional speech recognition or speaker identification tasks, for which either the identity of the speaker or the utterance information is unavailable. However, in the task of spoken LID/LR, both the identity of the speaker and the utterance information are not available, which makes an added challenge [2]. It plays a vital role in numerous multilingual speech processing applications as described in [1,2,15,20,21,30].

The spoken LID/LR systems can be broadly classified into two types, [25] namely explicit and implicit systems. Explicit spoken LID/LR systems use phoneme sequences derived from speech signals for recognition of language, whereas implicit systems use the derived language-specific speech features. The performance of explicit spoken LID/LR systems is better compared to implicit counterparts, which is achieved at the cost of an increase in the complexity. Implicit systems are of favored choice to many researchers for developing less complex and efficient spoken LID/LR systems.

The motivation for this work comes from the following facts: (1) In the context of Indian languages, few attempts have been reported in the field of spoken language recognition. One of the main reasons is due to the non-availability of standard native speech corpora covering majority of the Indian languages. (2) India is a multilingual nation having 22 official languages and 1650 unofficial languages [18]. These languages can be broadly classified into four major linguistic families, [15] namely (a) Indo-Aryan, (b) Dravidian, (c) Austroasiatic, and (d) Tibeto-Burman. The languages within the respective linguistic families are known to share some common set of scripts and phonemes, thereby exhibiting some similarities among them. Moreover, it is believed that Sanskrit (ancient Indian language) is the main root and many (not all) other Indian languages are evolved from it. The similarity between different languages poses significant challenges to develop spoken language recognition models for Indian languages. The implementation of explicit spoken LID/LR systems is practically not feasible due to the similarity issues among different Indian languages. As a result, the implicit systems become the only choice to proceed with the development of spoken LID/LR systems for Indian languages.

The main objectives of this paper are: (1) to design an implicit, less complex acoustic speech system for spoken language recognition in Indian languages using spectral features and (2) to design spoken LID/LR systems which can perform well even on shorter (10-s duration) speech utterances apart from longer (30-s or 45-s duration) speech utterances.

In this paper, harmonic sequences, named Fourier parameter (FP) features, are proposed to identify the language from the perceptual content of speech signals, instead of using the traditional spectral features. To the best of authors' knowledge, it is an early attempt to apply this new set of FP features, along with their associated first-order and

[1] The terms 'system,' 'model,' and 'classifier' are interchangeably used in this article.

second-order differences for the task of speaker-independent spoken language recognition. The FP features are evaluated on the Indian Institute of Technology Kharagpur Multilingual Indian Language Speech Corpus (IITKGP-MLILSC). For comparing the performance of FP features, the state-of-the-art legacy features such as the mel-frequency cepstral coefficients (MFCC) are also extracted from the speech signals of the said corpus.[2] Spoken LID/LR models are developed using support vector machines (SVM), artificial neural networks (ANN), and deep neural networks (DNN). Experimental results show that the proposed FP features are effective in recognizing different Indian languages and resulted in improving the recognition performance of the systems when compared to MFCC features. The recognition performance is further improved by combining MFCC and FP features. The performance of the proposed FP features is also evaluated on the Oriental Language Recognition Speech Corpus (AP18-OLR). Significant improvements in the performance of the spoken LID/LR systems using FP features, and the combination of MFCC and FP features are also observed with respect to this database.

The rest of the paper is organized as follows: Sect. 2 presents about the literature survey. Section 3 provides the brief details of two multilingual speech corpora used in this paper for performing the experimental study. Section 4 discusses about various topics, namely the theoretical aspects of the conventional block or frame-level processing for speech signals, FP and MFCC speech features for spoken LID/LR tasks, feature normalization, feature scaling, and finally the feature selection for dimensionality reduction. Section 5 presents the details of SVM, ANN, and DNN classifier architectures used in this paper to develop spoken LID/LR systems. Section 6 presents and discusses about the obtained experimental results. Section 7 concludes with the insights toward future extensions to this work.

2 Literature Review

A detailed literature survey on the state-of-the-art language identification, more specific toward speech features and models, is discussed by Ambikairajah et al. [2]. Most of the approaches employed spectral and prosodic features for spoken language recognition [6]. With respect to Indian languages, few attempts are reported in the area of spoken language recognition. Early attempts have been made on the recognition of Indian languages by Balleda et al. [5], using 17-dimensional MFCC feature vectors with vector quantization (VQ) for four Indian languages. Rao et al. [28] have explored prosodic features to develop language recognition models for four Indian languages. Leena et al. [19] have explored spectral features with auto-associative neural networks (AANN) for language recognition with varying duration of test speech samples for three Indian languages.

Maity et al. [21] have explored two spectral features, namely MFCC and linear predictive cepstral coefficients (LPCC) with Gaussian mixture models (GMM), to develop speaker-dependent and speaker-independent language recognition models for 27 Indian languages using IITKGP-MLILSC database. The corresponding recognition

[2] The terms ‘corpus’ and ‘database’ are interchangeably used in this article.

accuracies for both the cases are reported as 96% and 45%, respectively. Reddy et al. [30] have explored multilevel spectral and prosodic features with GMM, to develop language recognition models for 27 Indian languages using IITKGP-MLILSC database. The recognition accuracies reported using MFCC features and the combination of spectral and prosodic features are 51.42% and 62.13%, respectively. Nandi et al. [26] have explored magnitude and phase information of excitation source (represented by Hilbert envelope (HE) and residual phase (RP), respectively) present in the linear prediction (LP) residual signal with GMM, to develop language recognition models for 27 Indian languages using IITKGP-MLILSC database. The evidences of HE and RP from sub-segmental, segmental, and supra-segmental levels are combined in different ways to achieve language-specific excitation source information. The maximum recognition accuracy of 63.70% is reported with respect to these features.

Jothilakshmi et al. [15] have explored spectral features, namely MFCC and shifted delta cepstral (SDC) with hidden Markov models (HMM), GMM and ANN, to develop language recognition models for nine Indian languages. Koolagudi et al. [18] have explored two spectral features (MFCC and SDC) and a set of prosodic features (pitch contour, energy contour, zero-crossing rate, and duration) with ANN, to develop language recognition models for four Indian languages from Dravidian linguistic family. Mounika et al. [24] have explored MFCC features with DNN and DNN with attention (DNN-WA), to develop language recognition models for 12 Indian languages. Veera et al. [41] have explored residual cepstral coefficients (RCC), MFFC and SDC with DNN, DNN-WA, and the state-of-the-art i-vector systems, to develop language recognition models for 13 Indian languages. Improvement in the recognition performance is observed using DNN-WA model with combined RCC and MFCC features. Vuddagiri et al. [42] have explored MFCC features with i-vectors, DNN, and DNN-WA, to develop language recognition models for 23 Indian languages using International Institute of Information Technology Hyderabad—Indian Language Speech Corpus (IIITH-ILSC). It is observed that the performance of DNN-WA architecture is better than i-vector and DNN models. Bhanja et al. [7] have proposed new parameters to model the prosodic characteristics of the speech signal. The extracted prosodic features are combined with MFCC features to develop a two-stage LID system for seven northeast Indian languages. Three classifiers, namely ANN, GMM with universal background model (UBM), and i-vector-based SVM, have been used.

From the state-of the-art literature on spoken language recognition in Indian languages, it is observed that most of the works are focused on traditional spectral and prosodic features for capturing the language-specific information. To the best of authors' knowledge, none of the studies have analyzed the use of new spectral features for spoken language recognition in Indian languages. In this paper, new FP spectral features with their associated first-order and second-order differences are introduced. These new spectral features are used to develop spoken LID/LR models for 15 Indian languages using IITKGP-MLILSC database. The spoken LID/LR models are developed using the state-of-the-art classifiers, namely SVM, ANN, and DNN. Similar kind of LID/LR models is also developed for ten oriental languages using AP18-OLR database to evaluate the performance of the proposed FP spectral features for the task of spoken language recognition.

3 Multilingual Speech Corpora/Databases

In this paper, two multilingual speech databases, namely the Indian Institute of Technology Kharagpur Multilingual Indian Language Speech Corpus (IITKGP-MLILSC) [21] and the Oriental Language Recognition Speech Corpus (AP18-OLR), are used to develop spoken LID/LR systems in Indian and oriental languages, respectively. These databases are employed to develop and validate the spoken LID/LR systems in Indian and oriental languages using MFCC, FP, and combined MFCC + FP features. The details of these databases are summarized in the following sub-sections.

3.1 IITKGP-MLILSC Database

The IITKGP-MLILSC database was developed by the Indian Institute of Technology Kharagpur.[3] It comprises of recorded speech data in 27 major Indian languages, out of which 15 languages from 3 major Indian linguistic families, namely Indo-Aryan, Dravidian, and Tibeto-Burman, are considered in this paper for the task of speaker-independent spoken language recognition. Out of 15 chosen languages, 9 languages (Bengali, Chhattisgarhi, Gujarati, Hindi, Kashmiri, Punjabi, Rajasthani, Sanskrit, and Sindhi) are chosen from the Indo-Aryan family [30], 3 languages (Konkani, Tamil, and Telugu) are chosen from the Dravidian family [30], and 3 languages (Manipuri, Mizo, and Nagamese) are chosen from the Tibeto-Burman family [30]. On an average, each language in the database has around 5–10 min of speech recordings corresponding to at least ten speakers. More details of this database are provided in [21,30]. This database is freely available upon request, for non-commercial and academic research purpose.

3.2 AP18-OLR (AP16-OL7 + AP17-OL3) Multilingual Database

The AP18-OLR database [39] was developed to provide support for the oriental language recognition challenge[4] (AP18-OLR) organized by the center for speech and language technologies (CSLT) at Tsinghua University and SpeechOcean. It provides recorded speech data in 10 oriental languages which belong to 5 Asian linguistic families, namely Altaic, Austroasiatic, Austronesian, Indo-European, and Sino-Tibetan. Out of 10 languages, 4 languages (Japanese, Kazakh, Korean, and Uyghur) belong to Altaic family, and 3 languages (Cantonese, Mandarin, and Tibetan) belong to Sino-Tibetan family. The remaining 3 languages, namely Indonesian, Russian, and Vietnamese, belong to Austronesian, Indo-European, and Austroasiatic families, respectively. More details of this database are provided in [39] and can also be found in the challenge Web site.[5] This database is freely available upon request, for non-commercial and academic research purpose.

[3] http://www.iitkgp.ac.in.

[4] http://www.olrchallenge.org.

[5] http://cslt.riit.tsinghua.edu.cn/mediawiki/index.php/OLR_Challenge_2018.

The AP18-OLR database is a combination of two multilingual databases, namely AP16-OL7 and AP17-OL3. The AP16-OL7 [43] multilingual database was developed by the SpeechOcean.[6] It provides recorded speech data in 7 oriental languages (Cantonese, Indonesian, Japanese, Korean, Mandarin, Russian, and Vietnamese). On an average, each language in this database has around 10 h of speech recordings corresponding to 24 speakers (12 males and 12 females). The data set of each language was divided into an independent training and testing data sets, each containing recorded speech data of 18 and 6 independent speakers, respectively. This database has a variation in the recording environment with respect to languages. More details of this database are provided in [43] and can also be found in the challenge Web site.[7]

The AP17-OL3 [38] multilingual database was developed by NSFC[8]-M2ASR[9] project. It provides recorded speech data in 3 oriental languages (Kazakh, Tibetan, and Uyghur). On an average, each language in this database has around 10 h of speech recordings. Unlike AP16-OL7, this database has much more variations in terms of the recording environment and the number of speakers. More details of this database are provided in [38] and can also be found in the challenge Web site.[10]

4 Frame-Level Acoustic Speech Features for Spoken Language Recognition

4.1 Conventional Block Processing of Speech Signal

The method of conventional block processing (CBP) is used to extract the intrinsic segmental (frame level) and supra-segmental (across frames) acoustic features from speech signal. Prior to CBP, the speech signals are initially subjected to pre-processing, which includes low-pass filtering followed by pre-emphasis [36]. In CBP, the continuous speech signal is divided into a consecutive sequence of individual frames[11] (either in terms of overlapping or non-overlapping format) of short duration, and finally the segmental and supra-segmental features are extracted from them. The method of CBP can be mathematically described as follows:

Consider a discrete-time continuous speech signal, say $x(m)$ of finite duration t s, having sampling frequency F_s. Let C, Q, and R represent the type of the channel, bit resolution, and bit rate of the recorded speech signal, respectively. This speech signal is passed through a digital low-pass FIR filter, having the cutoff frequency $F_\mathrm{c}\ \left(F_\mathrm{c} < \frac{F_\mathrm{s}}{2}\right)$ [36]. The corresponding output is the desired low-pass filtered speech signal $x_\mathrm{f}(m)$, given by,

[6] http://en.speechocean.com.

[7] http://cslt.riit.tsinghua.edu.cn/mediawiki/index.php/OLR_Challenge_2016.

[8] National Natural Science Foundation of China, http://www.nsfc.gov.cn.

[9] Multilingual Minorlingual Automatic Speech Recognition, http://m2asr.cslt.org.

[10] http://cslt.riit.tsinghua.edu.cn/mediawiki/index.php/OLR_Challenge_2017.

[11] The terms ‘frame’ and ‘segment’ are interchangeably used in this article.

$$x_{\mathrm{f}}(m)=\sum_{n=0}^{P} h(n)x(m-n), \tag{1}$$

where P is the order of the low-pass filter, and h is the vector containing the filter coefficients.

The low-pass-filtered speech signal $x_{\mathrm{f}}(m)$ is then passed through a digital first-order pre-emphasis (high pass) filter to reduce the differences in the power levels of different frequency components present in the speech signal. The corresponding output is the pre-emphasized speech signal $x_{\mathrm{pe}}(m)$, given by,

$$x_{\mathrm{pe}}(m)=x_{\mathrm{f}}(m)-\alpha x_{\mathrm{f}}(m-1), \tag{2}$$

where α is the pre-emphasis constant.

The pre-emphasized speech signal $x_{\mathrm{pe}}(m)$ is then used in CBP, where it is segmented into l finite consecutive overlapping frames of short duration T_{f}, each having N_{f} samples. The corresponding segmented speech is represented in the form of a matrix x_{s}, given by,

$$[x_{\mathrm{s}}]=[s_1,\ s_2,\ldots,s_l], \tag{3}$$

where $s_1,\ s_2,\ldots,s_l$ denotes l vectors, each having dimension of $N_{\mathrm{f}}\times 1$, containing samples of respective speech segments. The matrix x_{s} has a dimension of $N_{\mathrm{f}}\times l$, indicating that the speech segments are arranged in columns and rows correspond to the individual frame samples.

The number of overlapping frames into which the given speech signal can be segmented, is computed using,

$$l=\left\lceil\left(1+\left(\frac{N_{\mathrm{s}}-N_{\mathrm{f}}}{N_{\mathrm{of}}}\right)\right)\right\rceil;\quad \ni\quad N_{\mathrm{f}}>N_{\mathrm{of}}\ \text{ or }\ T_{\mathrm{f}}>T_{\mathrm{of}}, \tag{4}$$

where N_{s} is the number of samples in the speech signal, N_{f} is the number of samples in individual frames, N_{of} is the number of new samples in individual frames after frame shift or overlap, T_{f} is the duration of the individual frame in ms, and T_{of} is the duration of the frame shift in ms. Here, N_{s} is the speech signal-dependent parameter, whereas N_{f} and N_{of} (or T_{f} and T_{of}) are tunable parameters, whose values are defined during the development phase of the spoken language recognition system. Equation (4) is expressed in terms of sample domain parameters for computing the number of overlapping frames.

The number of samples N_{s} available in the recorded speech signal having sampling frequency F_{s} and time duration t is given by,

$$N_{\mathrm{s}}=t\times F_{\mathrm{s}}. \tag{5}$$

The number of samples N_f in each individual frame l can be computed using,

$$N_\mathrm{f} = \frac{T_\mathrm{f}}{T_\mathrm{s}}; \quad \ni \quad T_\mathrm{f} > T_\mathrm{s}, \tag{6}$$

where T_s is the speech signal-dependent parameter which denotes the time duration of a single speech sample in ms, given by, $T_\mathrm{s} = \frac{1}{F_\mathrm{s}}$.

Overlapping the frame by shifting it with a duration of T_of is equivalent to shifting it by N_of samples, given by,

$$N_\mathrm{of} = \frac{T_\mathrm{of}}{T_\mathrm{s}}; \quad \ni \quad T_\mathrm{of} > T_\mathrm{s}. \tag{7}$$

Equation (4) can also be expressed in terms of time domain parameters by substituting (5), (6), and (7) for computing the number of overlapping frames.

The percentage of frame overlap F_{ol} can be computed using,

$$F_{ol} = \left(\frac{N_\mathrm{f} - N_\mathrm{of}}{N_\mathrm{f}} \right) \times 100\ \%. \tag{8}$$

It is evident from (8) that, if $F_{ol} = 50\%$, then $N_\mathrm{of} = \frac{N_\mathrm{f}}{2}$. Similarly, if $F_{ol} < 50\%$, then $N_\mathrm{of} < \frac{N_\mathrm{f}}{2}$, and if $F_{ol} > 50\%$, then $N_\mathrm{of} > \frac{N_\mathrm{f}}{2}$.

In speech signal segmentation, apart from the overlapping frames format, another format does exist for a special case having $F_{ol} = 0\%$ and it is termed as non-overlapping frames format. For this case, $N_\mathrm{of} = N_\mathrm{f}$ (or $T_\mathrm{of} = T_\mathrm{f}$), and the number of such frames into which the given speech signal can be segmented, is computed using the reduced form of (4), given by,

$$l = \left\lceil \frac{N_\mathrm{s}}{N_\mathrm{f}} \right\rceil. \tag{9}$$

The non-overlapping frames format for signal segmentation is rarely incorporated in speech signal analysis.

It is a usual practice to multiply the individual speech frames with a windowing function (whose window length N_w is equal to the frame length N_f) while segmenting the speech signal into a set of individual frames. The windowing operation helps to reduce the edge effects, while taking the discrete Fourier transform (DFT) on the speech segments [22]. The windowed speech segments are represented in the form of a matrix x_ws, given by,

$$[x_\mathrm{ws}] = [x_\mathrm{s}] \times w, \tag{10}$$

where w denotes a vector with windowing function coefficients having dimension of $N_w \times 1$ (here, $N_w = N_\mathrm{f}$), and × denotes the array multiplication. The matrix x_ws has a dimension of $N_\mathrm{f} \times l$ (similar to x_s), containing speech segments obtained after multiplying with the coefficients of the windowing function. In general, for speech applications, hamming window is widely used as a windowing function. It is defined as [27],

$$w(m) = 0.54 - 0.46 \cos\left(\frac{2\pi m}{K}\right); \quad for \;\; 0 \le m \le K,$$
$$K = N_w - 1 \;\; \& \;\; N_w = N_{\mathrm{f}}, \tag{11}$$

where K is the order of the filter, and N_w is the hamming window length.

The matrix x_{ws} finally contains the samples corresponding to the windowed speech segments. The segments with speech activity carry information of the language traits as opposed to the leading and trailing segments with silence or non-speech activity. Therefore, it becomes necessary to discard the unwanted leading and trailing silence or non-speech segments from x_{ws}. This is achieved by performing a simple voice activity detection (VAD) using segment energy estimation (SEE), which identifies the starting and ending boundaries of the entire speech utterance in the given speech signal. The process of VAD by SEE is briefly summarized as follows [2]:

Initially, the energy E_l of all the segments in x_{ws} is computed as,

$$E_l = 10\ \log_{10}\left(\sum_{m=0}^{N_{\mathrm{f}}-1} \left|x_{\mathrm{ws}}^{l}(m)\right|^2\right), \tag{12}$$

where E_l is the energy of lth frame in dB.

From energy estimates of E_l, the maximum energy $E_{\max}$ of the entire speech utterance is determined. Using $E_{\max}$, a threshold energy level E_{th} is computed as,

$$E_{\mathrm{th}} = \left(E_{\max} - E_{\mathrm{c}}\right), \tag{13}$$

where E_{c} is a tunable parameter which represents a constant energy in dB. It is used to adjust the level of E_{th}. Its value is defined during the development phase of the spoken language recognition system.

From (13), it can be noted that E_{th} is fixed at E_{c} dB below $E_{\max}$. Finally, all the leading and trailing speech segments whose energy fall below E_{th} are considered to be silence or non-speech segments and therefore discarded from x_{ws}. The resultant matrix obtained is denoted as $\widehat{x}_{\mathrm{ws}}$. This matrix has l' segments, where $l' < l$. The obtained speech segments in $\widehat{x}_{\mathrm{ws}}$ are then processed with the chosen speech feature extraction techniques to extract salient speech features that can be used to develop spoken LID/LR systems. The values of various CBP parameters used in this paper are presented in Table 1.

4.2 Fourier Parameter Features

A speech signal $x\,(m)$ which is divided into l consecutive overlapping frames can be represented by a combination of an FP model, given by [44],

Table 1 CBP parameters chosen with respect to multilingual speech databases

Type	Parameter	Value	
		For IITKGP-MLILSC	For AP18-OLR
Speech signal	t	10 s*	5–10 s
	F_s	8 kHz	16 kHz
	T_s	125 µs	62.5 µs
	C	Monochannel	Monochannel
	Q	16 bits/sample	16 bits/sample
	R	128 kbps	256 kbps
Low-pass filter	P	100	100
	F_c	3.4 kHz	6.8 kHz
Pre-emphasis filter	α	0.95	0.95
Speech segments	N_f	256	256
	N_of	128	128
	T_f	32 ms	16 ms
	T_of	16 ms	8 ms
	F_{ol}	50%	50%
Hamming window	K	255	255
	N_w	256	256
VAD by SEE	E_c	30 dB	30 dB

*For the purpose of experiments, each recorded speech wave file in IITKGP-MLILSC database is sliced into chunks of 10-s duration using WavePad® sound editor tool of NCH®

$$x(m) = \sum_{k=1}^{M} H_k^l(m)\left(\cos\left(2\pi \frac{f_k^l}{F_\mathrm{s}} m\right) + \phi_k^l\right), \tag{14}$$

where F_s is the sampling frequency of $x(m)$, H_k^l, f_k^l, and ϕ_k^l are the amplitude, frequency, and phase of the kth harmonic's sine component, respectively, l is the index of the frame, and M is the number of speech harmonic components.

The harmonic part of the model corresponds to the Fourier representation of the speech signal's periodic components. Since acoustic speech is non-periodic in nature, when its non-periodic components are sampled, the resultant Fourier transform becomes periodic and continuous function of frequency.

For a finite duration discrete-time speech signal $x(m)$ of length N samples, the DFT is defined as [27],

$$H(k) = \sum_{m=0}^{N-1} x(m) \exp\left(-j\frac{2\pi}{N}mk\right); \quad k = 0,\ 1,\ 2, \ldots, N-1, \tag{15}$$

where $H(k)$ are Fourier Parameters [44].

Harmonics generally include amplitude, frequency, and phase. In this paper, only the harmonic amplitudes are used as features. Harmonic amplitude FPs are estimated from each frame of the speech signal, as shown in (14), in which H_k^l is referred as lth frames FP. Intrinsic segmental and supra-segmental FPs are extracted from the speech segments (in $\widehat{x}_{\mathrm{ws}}$) to use them as features.

Initially, the characteristics of the harmonic amplitude FPs are studied by considering the mean statistical parameter. At first, one particular harmonic amplitude FPs are extracted from the frames of the speech signals corresponding to a single speaker from each language of both databases (IITKGP-MLILSC and AP18-OLR). Their corresponding means are computed across speech signals with respect to frames. The obtained results with respect to both databases (IITKGP-MLILSC and AP18-OLR) are presented in Figs. 1 and 2. These figures show the mean H_3 plots among languages in IITKGP-MLILSC and AP18-OLR databases, respectively. It is observed that the amplitudes vary with respect to different languages. The similar kind of variation is observed in the case of other harmonics. An adequate number of harmonic amplitude FPs are to be extracted from the speech signals to incorporate them for pattern classification/recognition problems, since it is difficult to classify/recognize signals based on the features obtained from single harmonics.

To investigate further in this direction, the first 120 harmonic amplitude FPs are extracted from a randomly chosen speech signal of each language for both databases (IITKGP-MLILSC and AP18-OLR). Their corresponding means are computed across the frames. The obtained results with respect to both databases (IITKGP-MLILSC and AP18-OLR) are presented in Figs. 3 and 4. Interesting characteristics can be studied from the maximum peaks of the mean harmonic amplitudes. For better peak visualization, scatter plots are provided separately in Figs. 5 and 6 for IITKGP-MLILSC and AP18-OLR databases, respectively. Figures 5 and 6 present the results of 6 random observation trails. Each observation trail (subplot) consists of the maximum peaks of the mean harmonic amplitudes corresponding to a randomly chosen speech signal from each language of the respective databases. From Fig. 5, it is observed that:

- For all observation trails, the majority of the Indian languages have the maximum peaks of the mean harmonic amplitudes at lower harmonics.
- In each observation trail, for any particular language, the maximum peak of the mean harmonic amplitude is formed at random harmonics. This is evident with the fact that for each observation trail, one speech signal is randomly picked from every language.
- To investigate this random nature of the peak formation, the maximum peaks of the mean harmonic amplitudes across adjacent harmonics and farthest maximum peaks of the mean harmonic amplitudes at the same harmonics are grouped together in the form of clusters (as indicated in the form of circles and ellipses in Fig. 5).
- Majority peaks of different languages within the cluster belong to one among the three major categories of Indian linguistic families. In Fig. 5, different cluster groups are denoted by different line styles. Clusters with solid, dashed, and dotted lines denote Indo-Aryan, Dravidian, and Tibeto-Burman families, respectively.
- For example, in Fig. 5c and 5f, three Dravidian languages, namely Konkani, Tamil, and Telugu, together formed a single cluster. Similarly, in Fig. 5d, e, two Dravidian languages, namely Konkani and Telugu, together formed a single cluster.

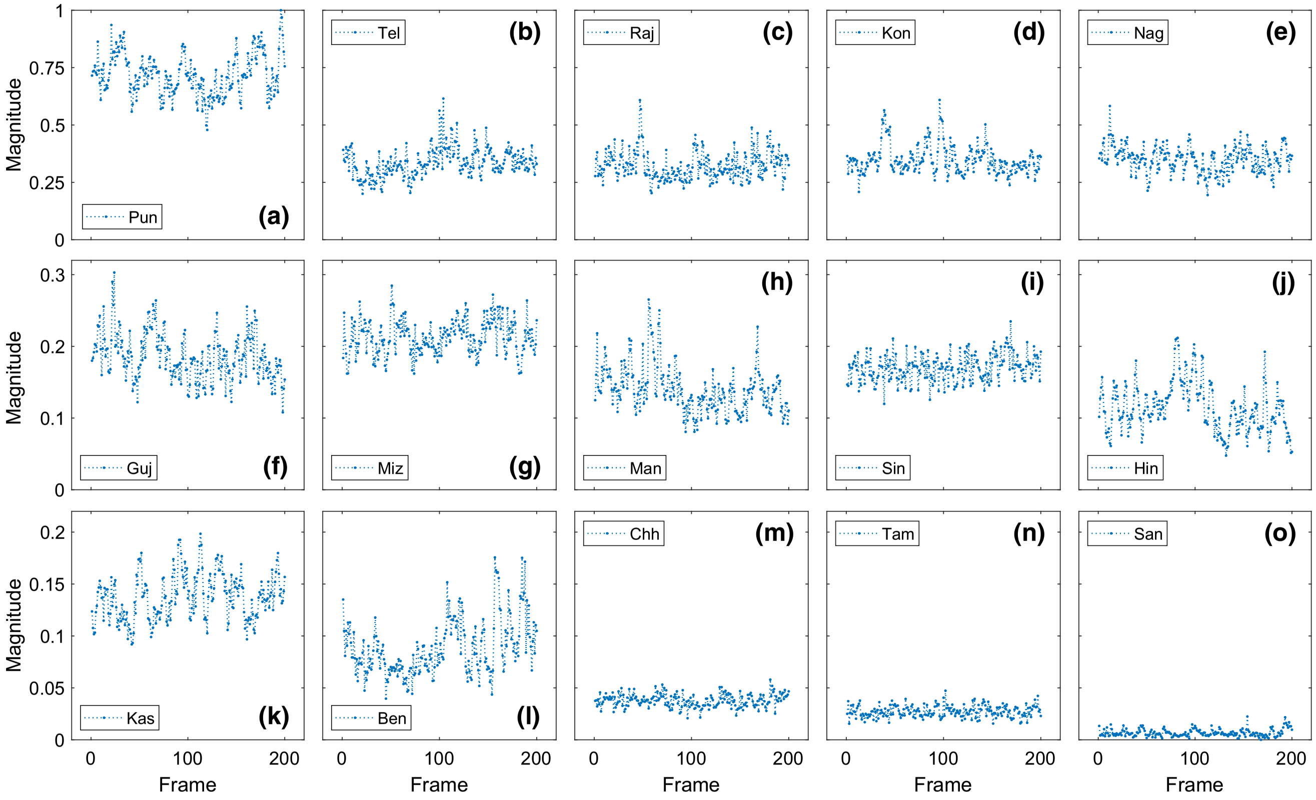

Fig. 1 Mean of H_3 for multilingual speech signals in IITKGP-MLILSC database corresponding to a single speaker per language. Here, the scale of y-axis is different for the sub-plots arranged in rows 1, 2, and 3. **a** Pun. = Punjabi. **b** Tel. = Telugu. **c** Raj. = Rajasthani. **d** Kon. = Konkani. **e** Nag. = Nagamese. **f** Guj. = Gujarati. **g** Miz. = Mizo. **h** Man. = Manipuri. **i** Sin. = Sindhi. **j** Hin. = Hindi. **k** Kas. = Kashmiri. **l** Ben. = Bengali. **m** Chh. = Chhattisgarhi. **n** Tam. = Tamil. **o** San. = Sanskrit

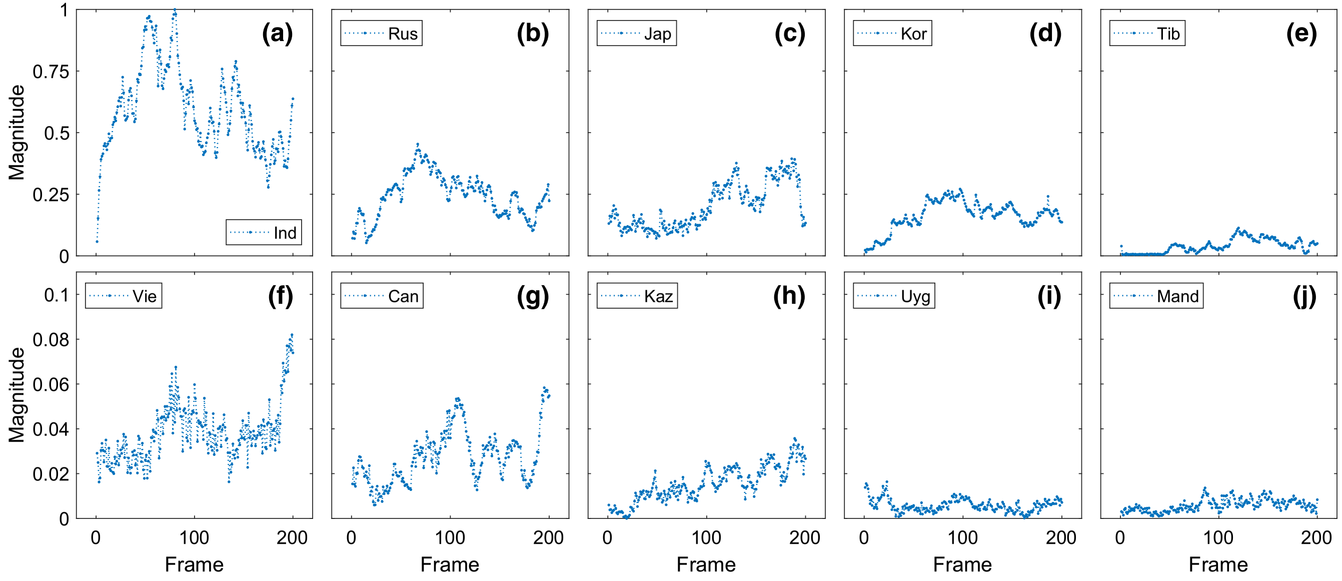

Fig. 2 Mean of H_3 for multilingual speech signals in AP18-OLR database corresponding to a single speaker per language. Here, the scale of y-axis is different for the sub-plots arranged in rows 1 and 2. **a** Ind. = Indonesian. **b** Rus. = Russian. **c** Jap. = Japanese. **d** Kor. = Korean. **e** Tib. = Tibetan. **f** Vie. = Vietnamese. **g** Can. = Cantonese. **h** Kaz. = Kazakh. **i** Uyg. = Uyghur. **j** Mand. = Mandarin

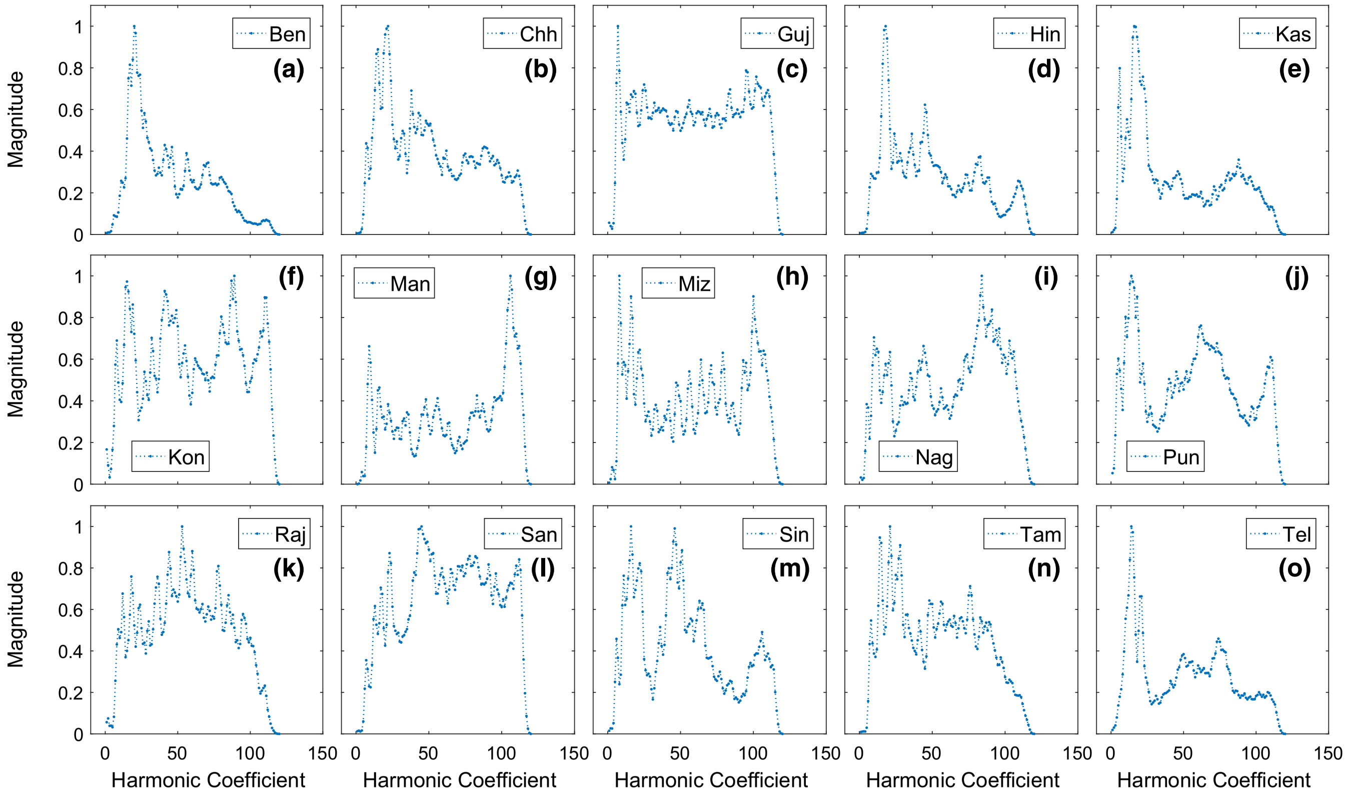

Fig. 3 Means of H_1 to H_{120} for multilingual speech signals in IITKGP-MLILSC database. **a** Ben. = Bengali. **b** Chh. = Chhattisgarhi. **c** Guj. = Gujarati. **d** Hin. = Hindi. **e** Kas. = Kashmiri. **f** Kon. = Konkani. **g** Man. = Manipuri. **h** Miz. = Mizo. **i** Nag. = Nagamese. **j** Pun. = Punjabi. **k** Raj. = Rajasthani. **l** San. = Sanskrit. **m** Sin. = Sindhi. **n** Tam. = Tamil. **o** Tel. = Telugu

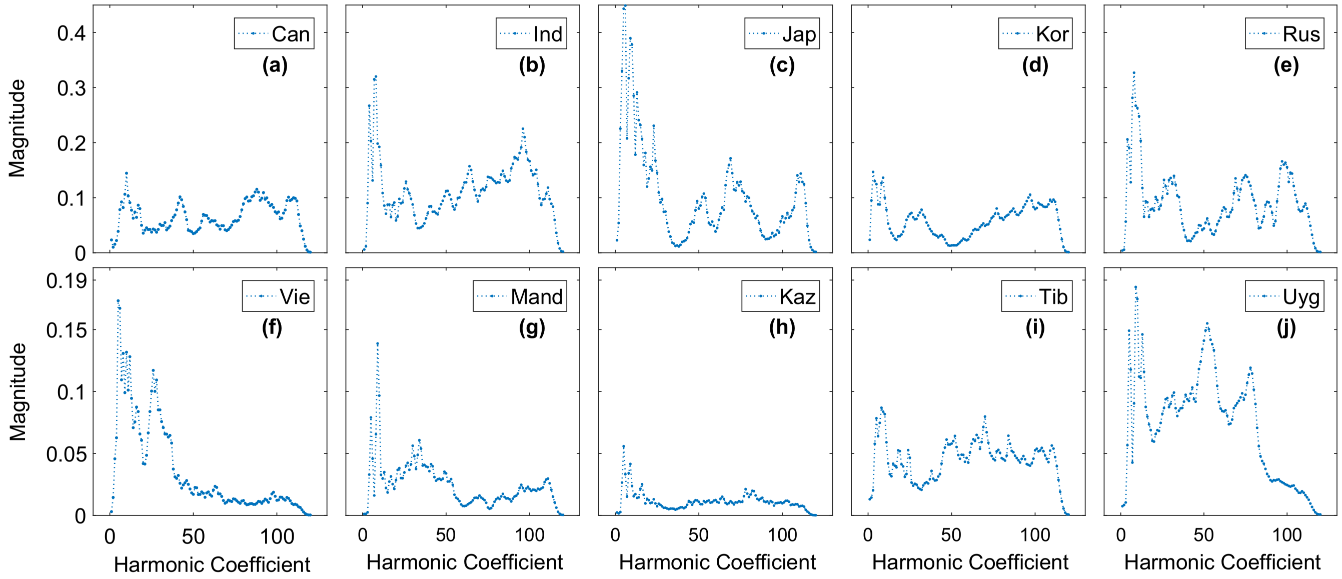

Fig. 4 Means of H_1 to H_{120} for multilingual speech signals in AP18-OLR database. Here, the scale of y-axis is different for the sub-plots arranged in rows 1 and 2. **a** Can. = Cantonese. **b** Ind. = Indonesian. **c** Jap. = Japanese. **d** Kor. = Korean. **e** Rus. = Russian. **f** Vie. = Vietnamese. **g** Mand. = Mandarin. **h** Kaz. = Kazakh. **i** Tib. = Tibetan. **j** Uyg. = Uyghur

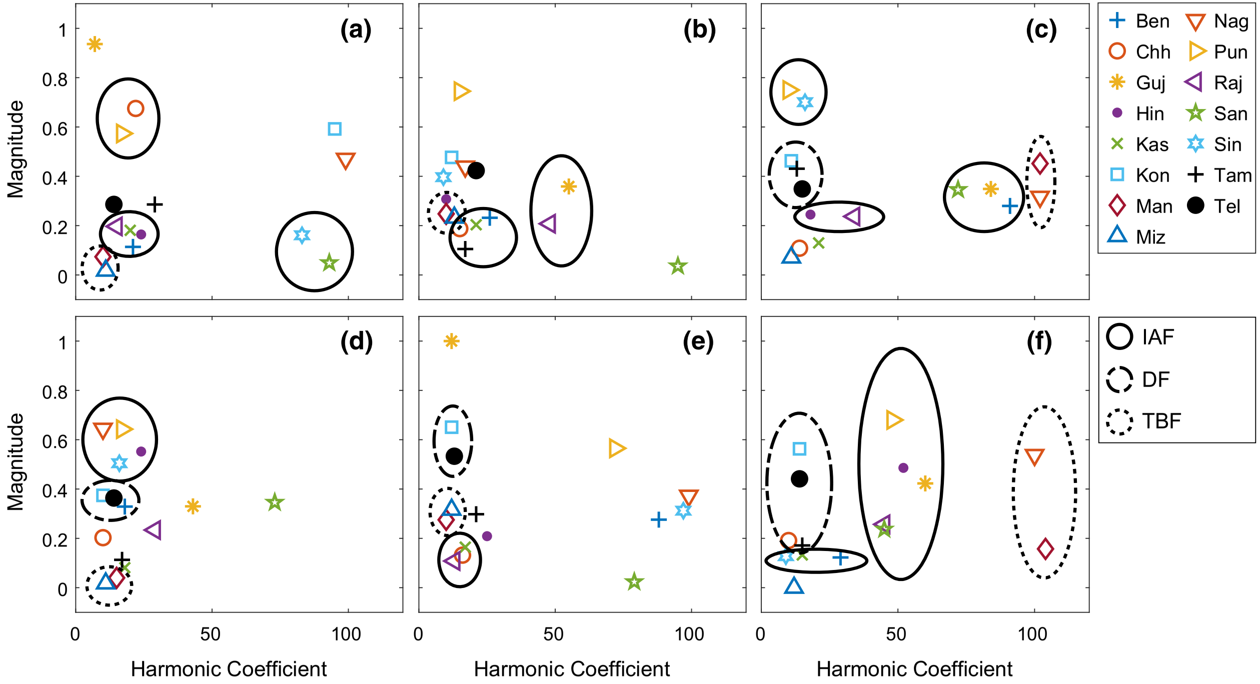

Fig. 5 Highest peaks corresponding to the means of H_1 to H_{120} for multilingual speech signals in IITKGP-MLILSC database. This plot shows the observations for six trails **a**–**f** performed by picking a random speech signal from each language. Ben. = Bengali, Chh. = Chhattisgarhi, Guj. = Gujarati, Hin. = Hindi, Kas. = Kashmiri, Kon. = Konkani, Man. = Manipuri, Miz. = Mizo, Nag. = Nagamese, Pun. = Punjabi, Raj. = Rajasthani, San. = Sanskrit, Sin. = Sindhi, Tam. = Tamil, and Tel. = Telugu. IAF. = Indo-Aryan Family, DF. = Dravidian Family, and TBF. = Tibeto-Burman Family

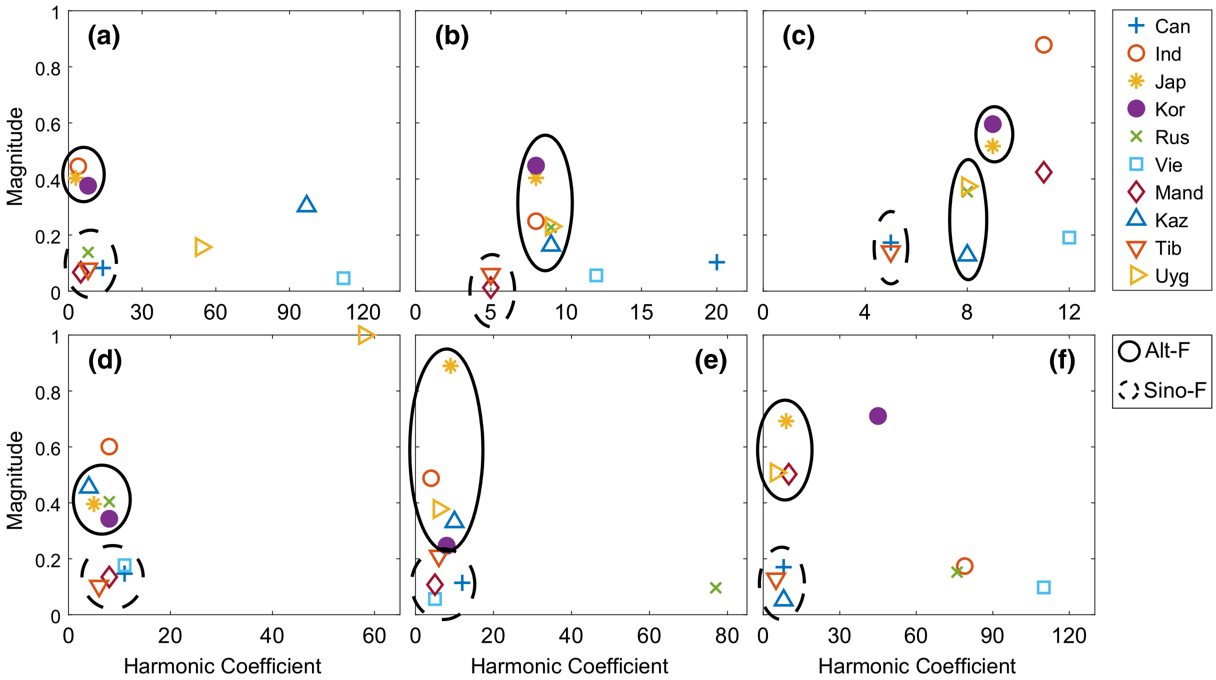

Fig. 6 Highest peaks corresponding to the means of H_1 to H_{120} for multilingual speech signals in AP18-OLR database. This plot shows the observations for six trails **a**–**f** performed by picking a random speech signal from each language. Here, the scale of x-axis is different for the sub-plots arranged in rows 1 and 2. Can. = Cantonese, Ind. = Indonesian, Jap. = Japanese, Kor. = Korean, Rus. = Russian, Vie. = Vietnamese, Mand. = Mandarin, Kaz. = Kazakh, Tib. = Tibetan, and Uyg. = Uyghur. Alt-F. = Altaic Family, and Sino-F. = Sino-Tibetan Family

- For example, in Fig. 5a, b, d, e, two Tibeto-Burman languages, namely Manipuri and Mizo, together formed a single cluster. Similarly, in Fig. 5c, f, two Tibeto-Burman languages, namely Manipuri and Nagamese, together formed a single cluster.
- Similar kind of analysis can also be made to Indo-Aryan languages as well, and the respective clusters so formed can be observed in Fig. 5.

Similarly, from Fig. 6, it is observed that:

- For all observation trails, majority of the oriental languages have the maximum peaks of the mean harmonic amplitudes at lower harmonics.
- The maximum peaks of the mean harmonic amplitudes are grouped together in the form of clusters using the same procedure followed in Fig. 5. Majority peaks of different languages within the cluster belong to one among the two categories of oriental linguistic families. In Fig. 6, different cluster groups are denoted by different line styles. Clusters with solid and dashed lines denote Altaic and Sino-Tibetan families, respectively.
- For example, in Fig. 6a, b, c, two Altaic languages, namely Japanese and Korean, together formed a single cluster. Similarly, in Fig. 6d, three Altaic languages, namely Japanese, Korean, and Kazakh, together formed a single cluster.
- For example, in Fig. 6a, d, e, three Sino-Tibetan languages, namely Cantonese, Mandarin, and Tibetan, together formed a single cluster. Similarly, in Fig. 6c, f, two Sino-Tibetan languages, namely Cantonese and Tibetan, together formed a single cluster.

The characteristics as described above are seen in multiple number of observation trails, where in each trail, a speech signal is randomly chosen from each language of the respective databases to generate similar kind of plots. The distinct characteristics exhibited by FPs show that there is a relationship associated with FPs and the language traits. This relationship is exploited to make use in the task of spoken language recognition. The characteristics of the harmonic amplitude FPs studied from Figs. 1, 2, 3, 4, 5, and 6 are with respect to the mean statistical parameter. A similar kind of investigation can be performed using other statistical parameters as well.

The global features usually provide superior performance capability in terms of the computational efficiency and classification accuracy [4]. Therefore, the statistical parameters like the mean, median, standard deviation, minimum, and maximum of the FP features across all l frames are calculated to derive global FP features. The computed global FP features are used to construct the global FP feature vectors. The resultant global FP feature vectors are used to develop spoken LID/LR systems.

The global FP feature vector is constructed for each speech signal as per the procedure described in [44], which can be briefly summarized as follows: At first, M set of FP features are extracted from all frames of the speech signal as described in (14). From each frame, the first 120 harmonic coefficients (H) are considered. Dynamic coefficients including 120 first-order difference (ΔH) and 120 second-order difference $(\Delta\Delta H)$ are computed. Finally the mean, median, standard deviation, minimum, and maximum of 120 harmonic amplitudes corresponding to (H_{1-120}), (ΔH_{1-120}), and $(\Delta\Delta H_{1-120})$ across all l frames are computed and are concatenated to form a 1800-dimensional global FP feature vector. The resultant structure of the feature vector is

Table 2 Structure of global FP feature vector

Feature index range	Feature description	Feature index range	Feature description	Feature index range	Feature description
1–120	$\bar{x}\ (H_k^p)$	601–720	$\bar{x}\ (\Delta H_k^p)$	1201–1320	$\bar{x}\ (\Delta\Delta H_k^p)$
121–240	$\tilde{x}\ (H_k^p)$	721–840	$\tilde{x}\ (\Delta H_k^p)$	1321–1440	$\tilde{x}\ (\Delta\Delta H_k^p)$
241–360	$\sigma_x\ (H_k^p)$	841–960	$\sigma_x\ (\Delta H_k^p)$	1441–1560	$\sigma_x\ (\Delta\Delta H_k^p)$
361–480	$\min\ (H_k^p)$	961–1080	$\min\ (\Delta H_k^p$	1561–1680	$\min\ (\Delta\Delta H_k^p)$
481–600	$\max\ (H_k^p)$	1081–1200	$\max\ (\Delta H_k^p)$	1681–1800	$\max\ (\Delta\Delta H_k^p)$

Here, the range of k is defined as, $1 \leq k \leq 120$, where k denotes the harmonic coefficients. The range of p is defined as $1 \leq p \leq l$, where l is the number of speech frames. H_k denotes FPs, ΔH_k denotes first-order FPs, and $\Delta\Delta H_k$ denotes second-order FPs. The statistical parameters are computed with respect to k across l. $\bar{x}. =$ mean, $\tilde{x}. =$ median, $\sigma_x. =$ standard deviation, $min. =$ minimum, and $\max. =$ maximum

depicted in Table 2. The global FP feature vectors of all speech signals, each having 1800 features, are finally used for the task of speaker-independent spoken language recognition.

4.3 Mel-Frequency Cepstral Coefficient Features

MFCC is considered as the benchmark feature set, widely employed in diverse fields of speech signal processing, since from the time they were first introduced in [10]. These features are popularly used for spoken language recognition. In this paper, MFCC features are used for performance comparison with the proposed FP features for the task of spoken language recognition. The mel-filter bank used for extracting MFCC features is based on MFCC-FB40 configuration. It comprises of 40 individual and equal height triangular filters with logarithmically spaced center frequencies. The filter bank spans across the desired frequency range of $(0, F_s/2]$ Hz. Further details of the mel-filter bank design equations and the procedure for MFCC feature extraction are provided in [29,37].

In this paper, global MFCC feature vectors are used for spoken language recognition. It includes the mean, median, standard deviation, minimum, and maximum of the traditional MFCC feature vector. At first, the 13-MFCCs along with their associated first-order difference (Δ-MFCC) and second-order difference ($\Delta\Delta$-MFCC) are extracted from the individual frames of speech signals to form a 39-dimensional feature vector [29]. Further, its mean, median, standard deviation, minimum, and maximum are computed across all frames to form a 195-dimensional global MFCC feature vector. The resultant structure of the feature vector is depicted in Table 3. The global MFCC feature vectors of all speech signals, each having 195 features, are finally used for the task of speaker-independent spoken language recognition.

4.4 Normalization and Scaling of Feature Vectors

Normalization and scaling are the two important data pre-processing techniques, widely used in the fields of data science and machine learning to standardize the

Table 3 Structure of global MFCC feature vector

Feature index range	Feature description
1–39	$\bar{x}$ (MFCC$_k^p$)
40–78	$\tilde{x}$ (MFCC$_k^p$)
79–117	σ_x (MFCC$_k^p$)
118–156	min (MFCC$_k^p$)
157–195	max (MFCC$_k^p$)

Here, the range of k is defined as, $1 \leq k \leq 39$, where k denotes the mel-frequency cepstral coefficients. The range of p is defined as $1 \leq p \leq l$, where l is the number of speech frames. MFCC$_k$ denotes mel-frequency cepstral coefficients. The statistical parameters are computed with respect to k across l

$\bar{x}.=$mean, $\tilde{x}.=$median, $\sigma_x.=$standard deviation, min. = minimum, and max. = maximum

data. These techniques are used in this paper to pre-process the data of feature vectors. These techniques assist to develop robust spoken LID/LR systems.

Feature normalization allows elimination of recording and speaker variability [8, 44], thereby preserving the effectiveness of language discrimination. In this paper, a simple mean-variance normalization is used for normalizing the feature vectors, which is given by,

$$f_j^i = \left(\frac{\hat{f}_j^i - \mu_j}{\sigma_j} \right). \tag{16}$$

Here, (16) normalizes each feature j of the feature vector i (from a given set of $i = 1, 2, \ldots, r$ feature vectors), using mean and variance of each feature j over all feature vectors, respectively. $\hat{f}_j^i$ is the feature j of the current feature vector i. Normalization does not bound the feature values to any specific range. It only makes the features to have a unit variance. So it is desirable to perform scaling after normalization, since most of the machine learning algorithms work better with scaled data (features).

Feature scaling makes the data of the feature vectors to fall within the specified range (say, $[0,\ 1]$ or $[-1,\ 1]$), which helps to improve the overall training (learning) efficiency of the classifier. The scaled feature vectors are usually fed as inputs to the classifier at the time of training and testing. In this paper, a simple min–max scaling is employed, which is defined as [11],

$$\begin{aligned} f_j^{'i} &= \left(\frac{(s_{\max} - s_{\min}) \times (f_j^i - \min(f_j))}{\left(\max(f_j) - \min(f_j)\right)} \right) + s_{\min}, \\ where\ f_j^{'i} &= \begin{cases} 0\ or\ -1, & \text{if } f_j^i = \min(f_j), \\ 1, & \text{if } f_j^i = \max(f_j), \\ (0, 1)\ or\ (-1, 1) & \text{otherwise.} \end{cases} \end{aligned} \tag{17}$$

Here, (17) scales the range of each feature j of the feature vector i (from a given set of $i = 1, 2, \ldots, r$ feature vectors) to the range specified by $[s_{\min}, s_{\max}]$, which is chosen either as $[0, 1]$ or $[-1, 1]$. $\min(f_j)$ and $\max(f_j)$ are the minimum and maximum values of feature j over all feature vectors, respectively. f_j^i is the feature j of the current feature vector i.

Feature scaling transform is applied only to the training data and not to the entire data set (including the test data set) [11]. The $\min(f_j)$ and $\max(f_j)$ must be preserved to use them for scaling: (1) the future inputs that will be applied to the classifier for performing additional training, (2) the new inputs that will be applied to the classifier for testing. Therefore, $\min(f_j)$ and $\max(f_j)$ effectively become an integral part of the classifier model, similar to its weights and biases.

Scaling the target[12] values are usually not necessary. If the original targets are scaled, then the classifier will be trained to produce outputs in the scaled range. So necessary post-processing tasks are to be made using $\min(t_j)$ and $\max(t_j)$ (where $\min(t_j)$ and $\max(t_j)$ are the minimum and maximum values of the target j over all target vectors, respectively) in order to convert the classifier outputs back to original targets. In such cases, $\min(t_j)$ and $\max(t_j)$ will also become an integral part of the classifier model, similar to $\min(f_j)$ and $\max(f_j)$. This is usually seen in the case of ANN classifiers, since by default they operate with the numeric targets.

4.5 Feature Selection

Processing the high-dimensional feature vectors requires huge computational resources and time [35]. Moreover, all the available features are not pertinent. The performance of the classification algorithm degrades with the presence of irrelevant, noisy, and redundant features. Therefore, it is necessary to reduce the dimensionality of the feature vectors to improve the efficiency and effectiveness of the classifiers. Thus, the method of feature selection is employed in this paper.

Feature selection is one of the traditional and the state-of-the-art dimensionality reduction methods which aim at finding a subset of useful features from the original feature vectors. It provides many benefits in terms of improving the understandability, scalability, generalization, and recognition capability of the classifiers [3]. In this paper, ReliefF algorithm[13] is employed to perform feature selection, whose details are briefly discussed in the following sub-section.

4.5.1 ReliefF Feature Selection

ReliefF algorithm belongs to the category of supervised and filter-method approach-based feature selection. It comes under the family of Relief-based feature selection algorithms (RFAs). It is capable of detecting the conditional dependencies between the feature vector attributes,[14] and provides a unified view on the attribute estimation

[12] The terms ‘target’ and ‘class’ are interchangeably used in this article.

[13] https://in.mathworks.com/help/stats/relieff.html.

[14] The terms ‘attribute’ and ‘feature’ are interchangeably used in this article.

for classification problems [32]. Attribute estimations involve computation of attribute scores (weights) which are used to rank and select top scoring features.

The locality of the estimates (predictor ranks and weights) is generally controlled by the user-defined parameter k.[15] If k is set to 1, then the computed estimates can become unreliable especially for the noisy data. If k is set to any positive integer, whose value is comparable with the total number of observations (instances), then ReliefF algorithm fails to find the important predictors. For most of the applications, the value of k can be safely set to 10 [32].

In this paper, the ReliefF algorithm is used to reduce the dimensionality of global FP and MFCC features and to improve the recognition accuracies of spoken LID/LR systems. The obtained results of ReliefF feature selection for global FP and MFCC features corresponding to IITKGP-MLILSC and AP18-OLR databases are shown in Figs. 7, 8, and 9. The plots of computed weights versus feature attributes for both global FP and MFCC features are shown in Figs. 7 and 8, respectively. The plots of computed weights (arranged in descending order of the magnitudes) versus assigned predictor ranks for both global FP and MFCC features are shown in Fig. 9. Selection of important features can be done either in terms of the estimated feature weights or assigned predictor ranks.

Figures 7 and 8 show the selection of important features in terms of the estimated feature weights. It is observed that for global FP and MFCC features, the estimated weights of all attributes vary through out the length of the feature vector. Attributes with relatively higher weights are considered to be significant, while those with relatively lower weights are considered to be insignificant. Significant features contribute more toward enhancing the recognition performance of the classifiers. So only the top attributes having relatively higher weights are selected, and the rest are ignored. For illustration purpose, Figs. 7 and 8 show the selection of top 50 features having relatively higher weights. From Fig. 7, it is observed that the ReliefF algorithm selects top 50 features among different sub-parts (features corresponding to H, ΔH, and $\Delta\Delta H$) of the feature vector. Figure 7 also depicts the importance of incorporating both first-order and second-order dynamic coefficients in global FP feature vectors by selecting significant amount of important features from them.

Figure 9 shows the selection of important features in terms of the assigned predictor ranks. Ranks are assigned to the attributes based on their estimated weights. The attribute with highest weight gets the lowest rank and vice versa. It is observed that as the feature rank increases, the corresponding feature weight decreases. So only the top attributes having relatively lower ranks are selected, and the rest are ignored. In either of the two ways, the selected attributes will always have relatively lower ranks and higher weights. Finally, the feature selection achieves the goal of dimensionality reduction in the case of global FP and MFCC feature vectors.

For the task of speaker-independent spoken LID/LR, the ReliefF feature selection algorithm is used to select top 900 and top 100 discriminative features from 1800 global FP and 195 global MFCC features, respectively.

[15] The user-defined parameter k in ReliefF feature selection refers to k-nearest neighbors. In this paper, the value of k is set to 10.

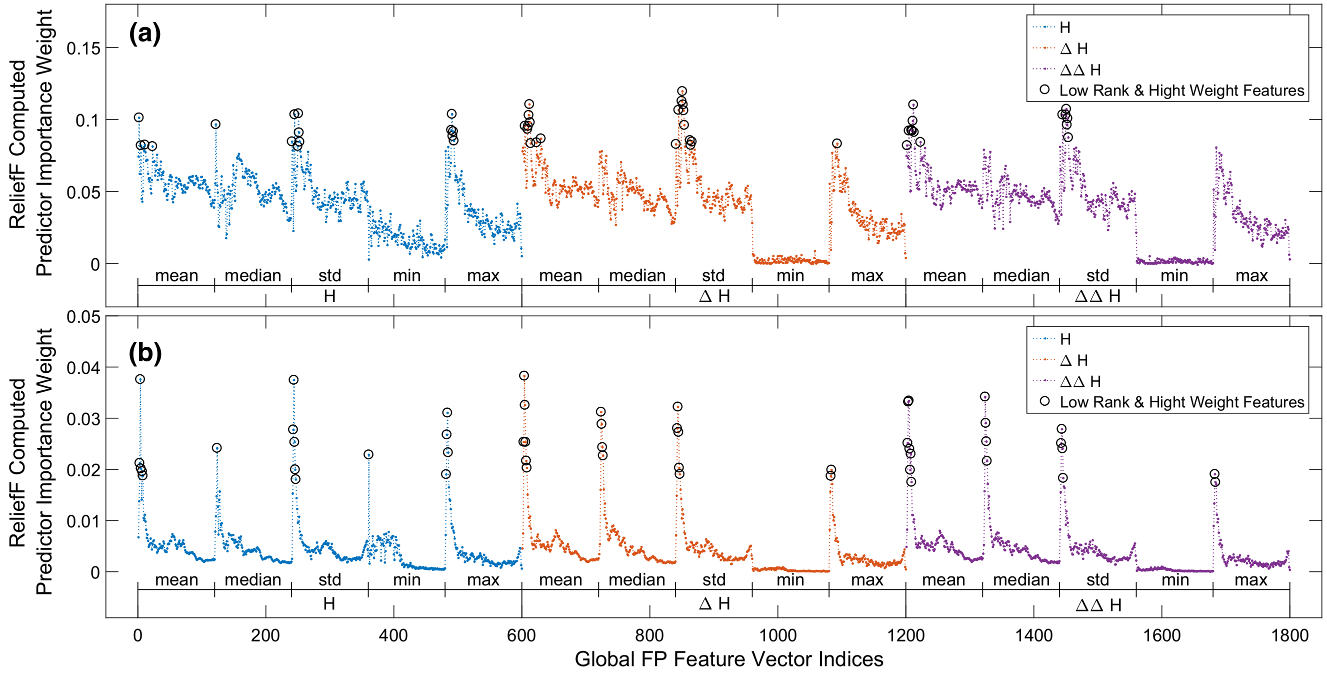

Fig. 7 ReliefF feature selection performed on global FP feature vectors of **a** IITKGP-MLILSC database and **b** AP18-OLR database. For illustration purpose, this figure shows the selection of top 50 features having low rank and high weight, as estimated using ReliefF feature selection algorithm. In the similar manner, top 900 features are selected from 1800 global FP features for the task of speaker-independent spoken language recognition

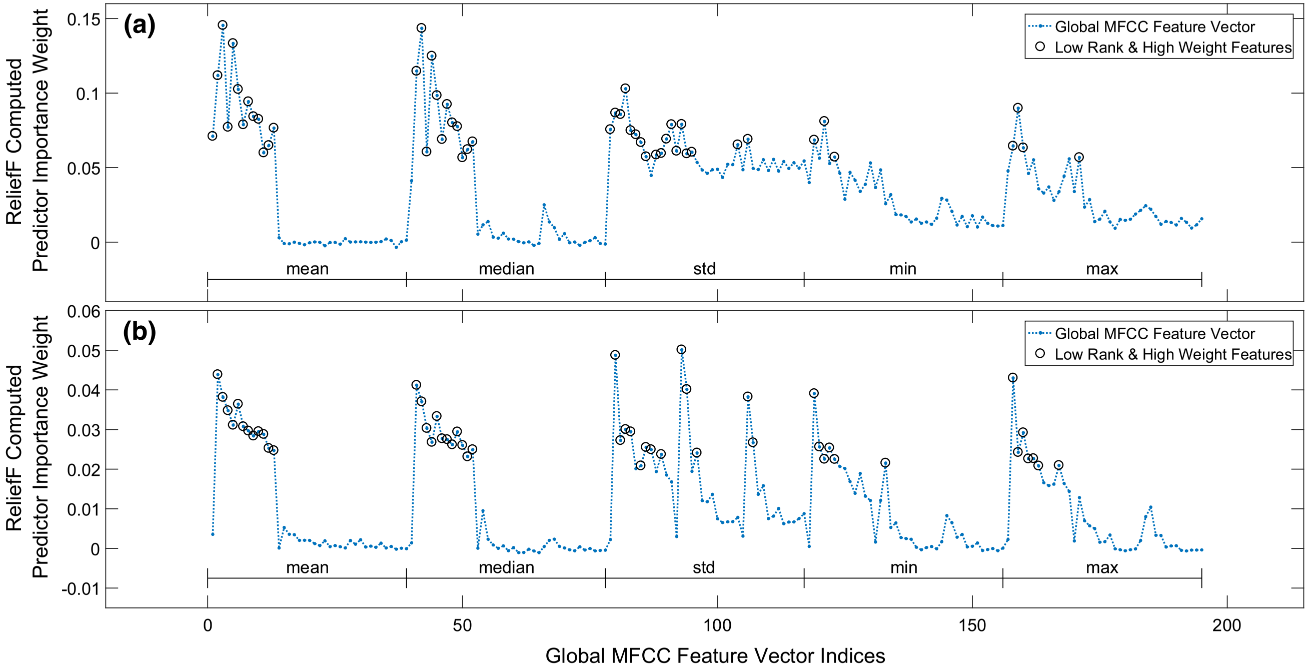

Fig. 8 ReliefF feature selection performed on global MFCC feature vectors of **a** IITKGP-MLILSC database and **b** AP18-OLR database. For illustration purpose, this figure shows the selection of top 50 features having low rank and high weight, as estimated using ReliefF feature selection algorithm. In the similar manner, top 100 features are selected from 195 global MFCC features for the task of speaker-independent spoken language recognition

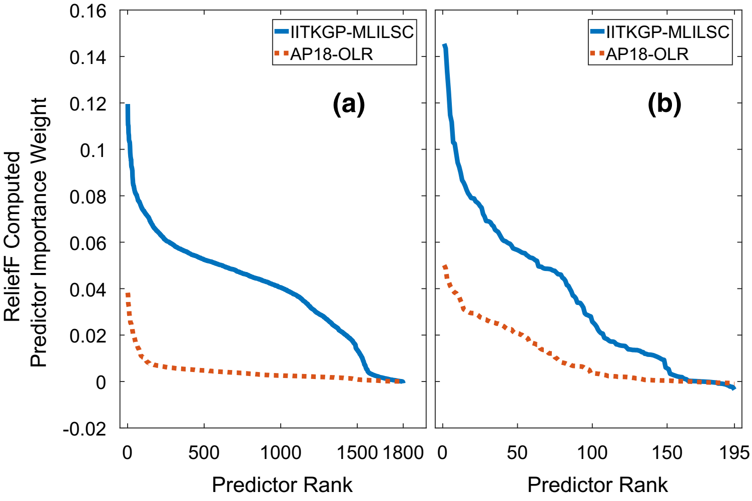

Fig. 9 Plot of ReliefF computed weights versus assigned ranks. **a** For global FP features. **b** For global MFCC features

5 Machine Learning Classification

This section discusses about the architectures of SVM, ANN, and DNN classifiers used in this paper to develop spoken LID/LR models in Indian and oriental languages.

5.1 Support Vector Machine Classification

In the literature, SVMs gained popularity by demonstrating good performance over classical problems in diverse fields of pattern recognition. They are widely incorporated due to the fact that they make use of the convex quadratic optimization which results in achieving a global optimal solution. They are discriminative in nature, and their performance is independent of the number of feature vectors [28]. They provide good generalization on classification problems by implementing the concept of structural risk minimization [40]. Originally, SVMs are designed for binary classification problems [9]. Different methods have been proposed in the literature for constructing a multiclass classifier by combining several binary classifiers [14].

Two such methods, namely 'one-versus-one' (OVO[16]) and 'one-versus-all' (OVA[17]) are considered in this paper to develop SVM-based spoken LID/LR models. Linear kernel function is chosen for SVM, since it is found to give good performance for spoken language recognition task [34]. Iterative single data algorithm (ISDA) is used as solver (for training the kernel machines). ISDA is ideal to carry out the learning process with huge training data sets, since it requires minimum computational resources [16].

[16] For a q-class classification problem, SVM with OVO configuration uses $\frac{q(q-1)}{2}$ binary learners.

[17] For a q-class classification problem, SVM with OVA configuration uses q binary learners.

In total, 2 different SVM architectures are considered to develop spoken LID/LR systems in Indian and oriental languages. The SVM models are separately trained and tested with global MFCC, FP, and the combination of MFCC + FP features. The obtained results are presented, analyzed, and discussed in Sect. 6.1.

5.2 Artificial Neural Network Classification

ANNs are capable in learning highly complex and nonlinear mappings between inputs and outputs. They are especially useful in applications where the underlying statistics of the considered task are not well known. There are wide varieties of neural network architectures available in the literature, out of which the architecture of multilayer feed-forward back-propagation is most commonly used in pattern classification/recognition problems. This architecture is considered in this paper to develop spoken language recognition models.

Three different variants of feed-forward back-propagation ANN architectures, each with one, two, and three hidden layers, respectively, are considered. With each variant, three further sub-variants are considered by choosing three different neuron activation functions, namely tan sigmod,[18] log sigmoid,[19] and elliot sigmoid,[20] respectively, in the hidden layers. Thus, a total of 9 different ANN architectures are used to develop spoken LID/LR systems. The neuron activation function in the output layer of all ANN models is kept fixed by considering softmax[21] function. The total number of passive neurons in the input layer and active neurons in the output layer is equal to the size of the feature vectors fed to the network as inputs and the number of output targets, respectively. The number of active neurons in the hidden layer(s) is set to $2/3$ times the number of passive neurons in the input layer (or the preceding hidden layer) [33] plus the number of active neurons in the output layer.

The feature vectors are divided into two different sets, namely training set and testing set. A small part of the training set is reserved as the validation set. ANN models are trained with the scaled conjugate gradient (SCG) back-propagation algorithm as described in [23]. The SCG algorithm is attractive for pattern recognition problems. It is very efficient and computationally faster than other training algorithms, for relatively larger networks having huge number of weights [13]. During the training (learning) phase, the SCG algorithm uses the training set to calculate the gradient and accordingly updates the network weights and biases. The training process is controlled by the validation parameter, named the cross-entropy error. It measures the network generalization and halts the training process when the network generalization stops improving (i.e., before over-fitting, which is indicated by an increase in the cross-entropy error). The cross-entropy error is evaluated on the validation set and is monitored during the training process. The testing set has no effect on the training and

[18] $\text{tansig}(n) = \frac{2}{(1+e^{-2n})-1}$.

[19] $\text{logsig}(n) = \frac{1}{(1+e^{-n})}$.

[20] $\text{elliotsig}(n) = \frac{0.5\,n}{1+|n|} + 0.5$.

[21] $\text{softmax}(n) = \frac{e^n}{\sum e^n}$.

validation process. It provides an independent measure of the network performance during and after training.

In total, 9 different ANN architectures are considered to develop spoken LID/LR systems in Indian and oriental languages. The ANN models are separately trained and tested with global MFCC, FP, and the combination of MFCC + FP features. The obtained results are presented, analyzed, and discussed in Sect. 6.2.

5.3 Deep Neural Network Classification

Recently, the splendid gains in the performance achieved using deep neural networks for classification problems have motivated the use of DNNs to develop spoken LID/LR systems [31]. There are wide varieties of DNN architectures available in the literature. The use of the traditional end-to-end DNNs poses a drawback to model spoken language recognition. In traditional end-to-end DNNs, the decision is usually taken at every frame and the context used is fixed, while the language clue is generally assigned to the whole speech utterance [24]. This paper overcomes the above-mentioned drawback by considering the long short-term memory networks-based recurrent neural networks (LSTM-RNN) as DNN (similar to the one used in [38]) to develop spoken LID/LR systems. The LSTM units perform utterance-wise classification by effectively capturing and memorizing the long temporal context [12].

In this paper, the LSTM-RNN network architecture is used to develop spoken LID/LR systems. This network is made up of 5 different types of layers, namely (in order) the sequence input layer, LSTM layer, fully connected (FC) layer, softmax layer, and classification output layer. The network starts with the sequence input layer, followed by the LSTM layer. The network ends with the FC, softmax, and classification output layers to make predictions about the language (targets or classes). The sequence input layer inputs the feature sequences into the network. The LSTM layer learns the long-term dependencies from the feature sequences. It performs additive interactions to improve gradient flow during the training process. In LSTM layers, the hyperbolic tangent[22] and sigmoid[23] functions are used as the state and gate activation functions, respectively. The FC layer is similar to the hidden layers in ANN. All hidden units (neurons) in the FC layer connect to all the hidden units in the previous layer. The network is made deeper by inserting additional (more than one) LSTM and FC layers. The number of hidden units in the last FC layer is equal to the number of targets. Optimal number of hidden units is considered in the LSTM and FC layers. The softmax layer applies a softmax function (also called as the normalized exponential) to the input. The softmax function is the output unit activation function after the last FC layer. Finally, the classification layer assigns each input to one of the k mutually exclusive classes using the cross-entropy function.

The training of LSTM-RNN networks is carried out using adaptive moment estimation (ADAM) algorithm [17]. The number of epoches is set between 500 and 2000. Three different mini-batch sizes, namely 32, 1024, and 2048, are considered. The training data are shuffled before each training epoch. The hyperparameters of the lay-

[22] $\tan h$.

[23] $\sigma(x) = \left(1 + e^{-x}\right)^{-1}$.

ers (LSTM and FC) and the network (apart from those mentioned in this paper) are set to default values.[24],[25],[26]

In total, 6 different LSTM-RNN network architectures, namely (1) a 5-layer network with 1 LSTM and 1 FC layer, (2) a 6-layer network with 1 LSTM and 2 FC layers, (3) a 6-layer network with 2 LSTM and 1 FC layers, (4) a 7-layer network with 1 LSTM and 3 FC layers, (5) a 7-layer network with 2 LSTM and 2 FC layers, and (6) a 8-layer network with 2 LSTM and 3 FC layers are considered to develop spoken LID/LR systems in Indian and oriental languages. The LSTM-RNN networks are separately trained and tested with global MFCC, FP, and the combination of MFCC + FP features. The obtained results are presented, analyzed, and discussed in Sect. 6.3.

6 Experimental Results and Discussion

The experimental results presented in this section are obtained by evaluating the spoken LID/LR models, developed using IITKGP-MLILSC and AP18-OLR databases. In the case of IITKGP-MLILSC database, the training and testing speech utterances have a fixed duration of 10 s, respectively. On an average, 23-min duration of speech data per language is used for training, while 10 min is used for testing. This accounts to an average of 4-min duration of speech data per speaker for training and 2 min for testing. In the case of AP18-OLR database, the entire training and development data sets are used for training and testing, respectively. The speakers and their corresponding feature vectors used for training and testing the models are completely independent and mutually exclusive for all the experiments presented in this paper. SVM, ANN, and DNN (LSTM-RNN) classifiers-based spoken LID/LR models are developed for Indian and oriental languages using global FP, MFCC, and the combination of MFCC + FP features.

Initially, the performance of FP features is evaluated by choosing 10, 20, 30, and 40 FP features (along with their associated first-order and second-order differences), for speaker-independent spoken LID/LR. The respective global FP feature vectors are constructed for different combinations of features, including (H), $(H + \Delta H)$, and $(H + \Delta H + \Delta\Delta H)$, in the similar manner as described in Sect. 4.2 and Table 2 (in the case of 120 FP features). Here, the only difference is in the total number of features present in the respective feature vectors. SVM and ANN classifiers are trained and tested with the resultant global FP feature vectors. The obtained results are presented in Fig. 10. From this figure, it is observed that the recognition accuracy increases with each increment of 10 FP features. Moreover, the recognition accuracy increases by incorporating the dynamic features, namely the first-order and second-order differences. On the other hand, the third-order and fourth-order differences are also evaluated and found to be ineffective toward improving the recognition accuracy. This figure clearly projects the significance of dynamic features in enhancing the recognition performance of the spoken LID/LR systems. The performance capability

[24] http://in.mathworks.com/help/deeplearning/ref/nnet.cnn.layer.lstmlayer.html.

[25] http://in.mathworks.com/help/deeplearning/ref/nnet.cnn.layer.fullyconnectedlayer.html.

[26] https://in.mathworks.com/help/deeplearning/ref/trainingoptions.html.

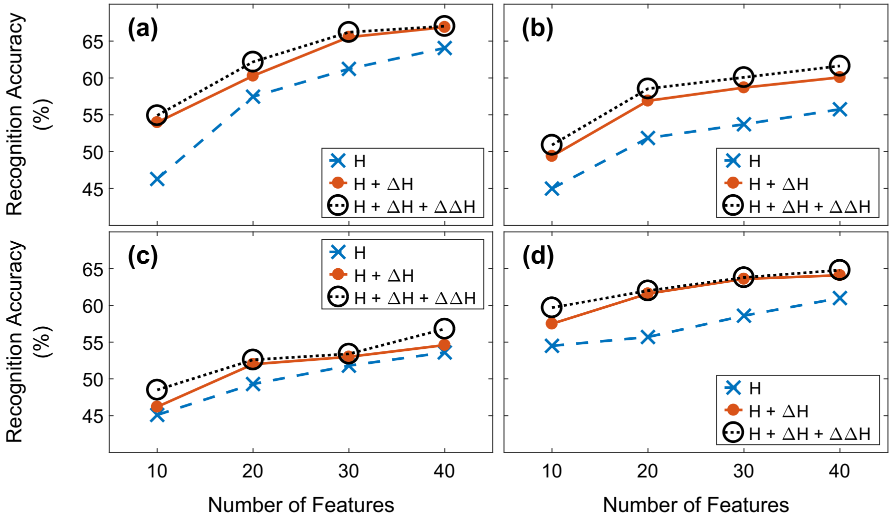

Fig. 10 Spoken language recognition using H, ΔH, and $\Delta\Delta H$. Here, x-axis represents the number of Fourier parameters. **a** Results of SVM classifier with respect to IITKGP-MLILSC database. **b** Results of SVM classifier with respect to AP18-OLR database. **c** Results of ANN classifier with respect to IITKGP-MLILSC database. **d** Results of ANN classifier with respect to AP18-OLR database

of these dynamic features has resulted to take them into account while constructing global FP feature vectors, as described in Sect. 4.2 and Table 2.

Finally, the performance of 120 FP features is evaluated by developing spoken LID/LR systems using SVM, ANN, and DNN (LSTM-RNN) classifiers. The performance of global FP features is then compared with the performance of global MFCC features. The net effect in the performance of spoken LID/LR systems using the combination of global MFCC + FP features is also evaluated. The obtained results are independently analyzed with respect to SVM, ANN, and DNN (LSTM-RNN) classifiers in Sects. 6.1, 6.2, and 6.3, respectively. The best results achieved by SVM, ANN, and DNN (LSTM-RNN) classifiers using global MFCC, FP, and the combination of MFCC + FP features are compared in Sect. 6.4.

6.1 SVM-Based Spoken LID/LR Systems

Table 4 presents the results of recognition accuracies achieved by SVM classifiers using global MFCC, FP, and MFCC + FP features on IITKGP-MLILSC and AP18-OLR databases. The maximum recognition accuracies achieved with respect to different feature sets on both databases are marked in bold. As can be seen from Table 4, the models with OVA configuration perform well when compared to OVO configuration. For example, in the case of MFCC + FP features on IITKGP-MLILSC database, the OVA configuration achieves the recognition accuracy of 86.40%. On the other hand, for the same feature set, the OVO configuration achieves the recognition accuracy of 78.40%, showing a significant reduction in the performance. Comparatively, the

Table 4 Recognition performance of SVM-based spoken LID/LR systems

Features	IITKGP-MLILSC		AP18-OLR	
	OVO-SVM	OVA-SVM	OVO-SVM	OVA-SVM
MFCC	**67.70**	64.80	57.62	**59.16**
FP	68.00	**73.40**	62.63	**62.83**
MFCC + FP	78.40	**86.40**	69.67	**70.73**

Bold values indicate the maximum recognition accuracies of SVM-based spoken LID/LR systems with respect to databases and feature sets

models with OVA configuration are less complex and computationally efficient over the models with OVO configuration with respect to the number of binary learners.

In the case of IITKGP-MLILSC database, with respect to the proposed FP features, the SVM model with OVA configuration achieves the maximum recognition accuracy of 73.40%. The corresponding confusion matrix is shown in Table 5.

As can be seen from Table 5, languages, namely Bengali, Hindi, Manipuri, Tamil, and Telugu, are majorly mis-classified with other languages. Similarly, with respect to MFCC features, the SVM model with OVO configuration achieves the maximum recognition accuracy of 67.70%. The use of FP features shows an improvement[27] in the recognition accuracy by 8.42% when compared to MFCC features. With respect to MFCC + FP features, the SVM model with OVA configuration achieves the maximum recognition accuracy of 86.40%. The use of MFCC + FP features shows a significant improvement in the recognition accuracies by 27.62% and 17.71% when compared to MFCC and FP features, respectively. Finally, the recognition accuracies achieved with respect to individual languages by the best SVM models trained with global MFCC, FP, and MFCC + FP features on IITKGP-MLILSC database are depicted in a bar graph comparison plot as shown in Fig. 11. It is clear from Fig. 11 that the use of combined MFCC + FP features outperforms the use of MFCC and FP features, for most of the languages.

In the case of AP18-OLR database, with respect to the proposed FP features, the SVM model with OVA configuration achieves the maximum recognition accuracy of 62.83%. The corresponding confusion matrix is shown in Table 6. As can be seen from Table 6, languages, namely Indonesian, Korean, Mandarin, and Kazakh, are majorly mis-classified with other languages. Similarly, with respect to MFCC features, the SVM model with OVA configuration achieves the maximum recognition accuracy of 59.16%. The use of FP features shows an improvement in the recognition accuracy by 6.20% when compared to MFCC features. With respect to MFCC + FP features, the SVM model with OVA configuration achieves the maximum recognition accuracy of 70.73%. The use of MFCC + FP features shows a significant improvement in the recognition accuracies by 19.56% and 12.57% when compared to MFCC and FP features, respectively. Finally, the recognition accuracies achieved with respect to individual languages by the best SVM models trained with global MFCC, FP, and MFCC + FP features on AP18-OLR database are depicted in a bar graph com-

[27] The percentage improvement $I_{\%}$ in recognition accuracy is computed using, $I_{\%} = \frac{A_1 - A_2}{A_2} \times 100\% \ni A_1 > A_2$, where A_1 and A_2 are recognition accuracies in percentages.

Table 5 Confusion matrix of SVM classifier for speaker-independent spoken language recognition using 120 FP features on IITKGP-MLILSC database (%)

	Ben.	Chh.	Guj.	Hin.	Kas.	Kon.	Man.	Miz.	Nag.	Pun.	Raj.	San.	Sin.	Tam.	Tel.
Ben.	**28.57**			35.72	7.14									28.57	
Chh.		**100.00**													
Guj.			**90.38**										9.62		
Hin.	5.88			**47.06**	32.35									14.71	
Kas.				29.41	**67.65**									2.94	
Kon.	3.45		1.72			**91.38**							3.45		
Man.				13.33	8.33		**41.67**		3.34				20.00	13.33	
Miz.								**100.00**							
Nag.			1.69						**98.31**						
Pun.				29.42	11.76		2.94	2.94		**50.00**			2.94		
Raj.				3.33							**96.67**				
San.	5.56											**94.44**			
Sin.		1.66		3.33	6.67				6.67				**76.67**	5.00	
Tam.	26.67			30.00										**43.33**	
Tel.				14.72				5.88		8.82			26.47	11.76	**32.35**

The rows and columns of the confusion matrix correspond to the targets and outputs, respectively. The diagonal elements are marked in bold, indicating the percentage of true classification with respect to individual languages. The off-diagonal elements indicate the percentage of mis-classification

Ben. = Bengali, Chh. = Chhattisgarhi, Guj. = Gujarati, Hin. = Hindi, Kas. = Kashmiri, Kon. = Konkani, Man. = Manipuri, Miz. = Mizo, Nag. = Nagamese, Pun. = Punjabi, Raj. = Rajasthani, San. = Sanskrit, Sin. = Sindhi, Tam. = Tamil, and Tel. = Telugu

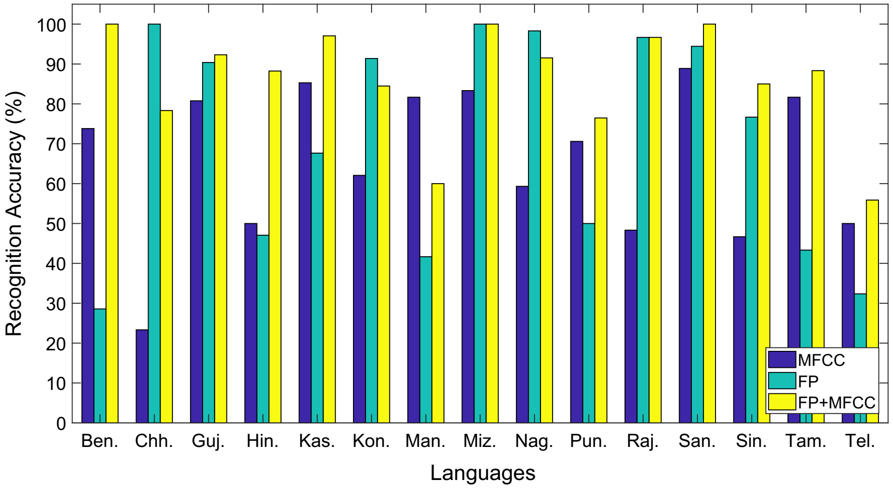

Fig. 11 Comparison of individual language recognition accuracies of best SVM classifiers with respect to global MFCC, FP, and MFCC + FP features on IITKGP-MLILSC database. The results depicted in this plot correspond to the models presented in Table 4, whose recognition accuracies are marked in bold. Ben. = Bengali, Chh. = Chhattisgarhi, Guj. = Gujarati, Hin. = Hindi, Kas. = Kashmiri, Kon. = Konkani, Man. = Manipuri, Miz. = Mizo, Nag. = Nagamese, Pun. = Punjabi, Raj. = Rajasthani, San. = Sanskrit, Sin. = Sindhi, Tam. = Tamil, and Tel. = Telugu

Table 6 Confusion matrix of SVM classifier for speaker-independent spoken language recognition using 120 FP features on AP18-OLR database (%)

	Can.	Ind.	Jap.	Kor.	Rus.	Vie.	Mand.	Kaz.	Tib.	Uyg.
Can.	**82.79**	1.12	3.23	1.56	2.30	1.04	0.06	6.31	0.48	1.11
Ind.	0.37	**33.14**		7.25	7.30	7.36	35.18	7.36		2.04
Jap.	0.05	14.55	**73.03**	4.85	0.16		0.26	0.26	6.21	0.63
Kor.	12.17	5.11	11.28	**32.42**	4.95	5.78	8.56	13.45	5.45	0.83
Rus.	0.06	3.63	0.39	9.98	**65.59**	6.53	8.80	1.62	1.45	1.95
Vie.	1.06	0.28	0.45	2.38	9.07	**85.09**	0.83	0.67	0.17	
Mand.	22.50	3.78	1.00	6.00	4.89	8.28	**49.44**	1.56	0.28	2.27
Kaz.	4.33	1.18	1.88	2.78	10.50	5.22	22.67	**32.00**	5.22	14.22
Tib.	0.17		0.11	0.11	0.11		0.22		**99.00**	0.28
Uyg.	3.57	0.20	0.62	1.18	1.47	0.20	2.26	2.18	9.09	**79.23**

The rows and columns of the confusion matrix correspond to the targets and outputs, respectively. The diagonal elements are marked in bold, indicating the percentage of true classification with respect to individual languages. The off-diagonal elements indicate the percentage of mis-classification
Can. = Cantonese, Ind. = Indonesian, Jap. = Japanese, Kor. = Korean, Rus. = Russian, Vie. = Vietnamese, Mand. = Mandarin, Kaz. = Kazakh, Tib. = Tibetan, and Uyg. = Uyghur

parison plot as shown in Fig. 12. It is clear from Fig. 12 that the use of combined MFCC + FP features outperforms the use of MFCC and FP features, for most of the languages.

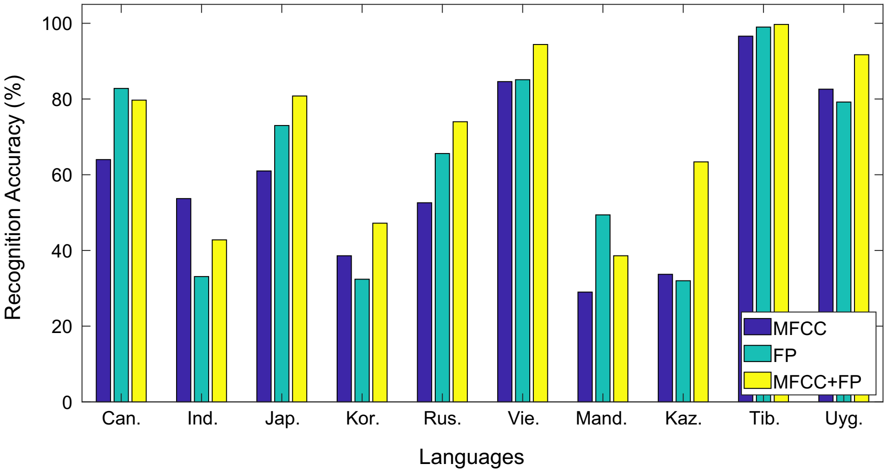

Fig. 12 Comparison of individual language recognition accuracies of best SVM classifiers with respect to global MFCC, FP, and MFCC + FP features on AP18-OLR database. The results depicted in this plot correspond to the models presented in Table 4, whose recognition accuracies are marked in bold. Can. = Cantonese, Ind. = Indonesian, Jap. = Japanese, Kor. = Korean, Rus. = Russian, Vie. = Vietnamese, Mand. = Mandarin, Kaz. = Kazakh, Tib. = Tibetan, and Uyg. = Uyghur

6.2 ANN-Based Spoken LID/LR Systems

Table 7 presents the results of recognition accuracies achieved by ANN classifiers using global MFCC, FP, and MFCC + FP features on IITKGP-MLILSC and AP18-OLR databases. The maximum recognition accuracies achieved with respect to different feature sets on both databases are marked in bold. As can be seen from Table 7, there is a significant drop in the recognition performance with respect to an increase in the number of hidden layers, for most of the cases. For example, in the case of MFCC + FP features on IITKGP-MLILSC database, the ANN model with 1 hidden layer using log-sigmoid activation function achieves the recognition accuracy of 89.40%. On the other hand, for the same set of features and activation function, the corresponding ANN models with 2 and 3 hidden layers achieve the recognition accuracies of 88.60% and 88.30%, respectively, showing a nominal drop in the performance. The use of a single hidden layer is found to be sufficient for achieving reasonably good recognition performance. Comparatively, the models with single hidden layer are less complex and computationally efficient over the models with two and three hidden layers. It is observed that most of the models with tan-sigmoid activation function in the hidden layers perform well when compared to the rest.

In the case of IITKGP-MLILSC database, with respect to the proposed FP features, the ANN model with 1 hidden layer using tan-sigmoid activation function achieves the maximum recognition accuracy of 74.20%. The corresponding confusion matrix is shown in Table 8. As can be seen from Table 8, languages, namely Bengali, Hindi, and Punjabi, are majorly mis-classified with other languages. Similarly, with respect to MFCC features, the ANN models with 2 and 3 hidden layers using tan-sigmoid activation function achieve the maximum recognition accuracy of 71.10%, respec-

Table 7 Recognition performance of ANN-based spoken LID/LR systems

Feature	Hidden layer activation function	IITKGP-MLILSC			AP18-OLR		
		Number of hidden layers			Number of hidden layers		
		1	2	3	1	2	3
MFCC	Tan sigmoid	70.00	**71.10**	**71.10**	59.50	59.30	60.80
	Log sigmoid	70.50	70.30	70.90	**61.30**	59.70	60.20
	Elliot sigmoid	69.40	69.00	65.70	58.80	57.40	56.80
FP	Tan sigmoid	**74.20**	72.90	71.00	**64.70**	64.00	**64.70**
	Log sigmoid	72.50	69.30	70.80	63.80	64.20	**64.70**
	Elliot sigmoid	71.50	69.40	69.90	62.20	62.90	56.00
MFCC + FP	Tan sigmoid	89.00	86.90	88.70	70.50	70.40	64.70
	Log sigmoid	**89.40**	88.60	88.30	64.80	**70.80**	**70.80**
	Elliot sigmoid	86.50	85.60	84.80	70.50	70.70	70.30

Bold values indicate the maximum recognition accuracies of ANN-based spoken LID/LR systems with respect to databases and feature sets

tively. The use of FP features shows an improvement in the recognition accuracy by 4.36% when compared to MFCC features. With respect to MFCC + FP features, the ANN model with 1 hidden layer using log-sigmoid activation function achieves the maximum recognition accuracy of 89.40%. The use of MFCC + FP features shows a significant improvement in the recognition accuracies by 25.74% and 20.49% when compared to MFCC and FP features, respectively. Finally, the recognition accuracies achieved with respect to individual languages by the best ANN models trained with global MFCC, FP, and MFCC + FP features on IITKGP-MLILSC database are depicted in a bar graph comparison plot as shown in Fig. 13. It is clear from Fig. 13 that the use of combined MFCC + FP features outperforms the use of MFCC and FP features, for most of the languages.

In the case of AP18-OLR database, with respect to the proposed FP features, the ANN models with 1 and 3 hidden layers using tan-sigmoid and log-sigmoid activation functions achieve the maximum recognition accuracy of 64.70%, respectively. The corresponding confusion matrix is shown in Table 9. As can be seen from Table 9, languages, namely Russian, Mandarin, and Uyghur, are majorly mis-classified with other languages. Similarly, with respect to MFCC features, the ANN model with 1 hidden layer using log-sigmoid activation function achieves the maximum recognition accuracy of 61.30%. The use of FP features shows an improvement in the recognition accuracy by 5.55% when compared to MFCC features. With respect to MFCC + FP features, the ANN models with 2 and 3 hidden layers using log-sigmoid activation function achieve the maximum recognition accuracy of 70.80%, respectively. The use of MFCC + FP features shows a significant improvement in the recognition accuracies by 15.50% and 9.43% when compared to MFCC and FP features, respectively. Finally, the recognition accuracies achieved with respect to individual languages by the best ANN models trained with global MFCC, FP, and MFCC + FP features on AP18-OLR

Table 8 Confusion matrix of ANN classifier for speaker-independent spoken language recognition using 120 FP features on IITKGP-MLILSC database (%)

	Ben.	Chh.	Guj.	Hin.	Kas.	Kon.	Man.	Miz.	Nag.	Pun.	Raj.	San.	Sin.	Tam.	Tel.
Ben.	**21.74**	2.17		21.74	23.92									30.43	
Chh.		**100.00**													
Guj.			**80.77**										19.23		
Hin.				**47.06**	20.59									32.35	
Kas.				35.29	**61.76**									2.95	
Kon.			8.62			**81.03**							10.35		
Man.				1.72	6.90		**81.03**						10.35		
Miz.								**100.00**							
Nag.			10.17						**89.83**						
Pun.				20.59	14.71		17.65	2.94	2.94	**23.53**			11.76		5.88
Raj.		3.34									**83.33**			13.33	
San.	22.22											**77.78**			
Sin.		5.00			1.67		10.00		5.00				**76.66**	1.67	
Tam.	20.00			23.33										**56.67**	
Tel.					2.94			2.94					2.94	5.89	**85.29**

The rows and columns of the confusion matrix correspond to the targets and outputs, respectively. The diagonal elements are marked in bold, indicating the percentage of true classification with respect to individual languages. The off-diagonal elements indicate the percentage of mis-classification

Ben. = Bengali, Chh. = Chhattisgarhi, Guj. = Gujarati, Hin. = Hindi, Kas. = Kashmiri, Kon. = Konkani, Man. = Manipuri, Miz. = Mizo, Nag. = Nagamese, Pun. = Punjabi, Raj. = Rajasthani, San. = Sanskrit, Sin. = Sindhi, Tam. = Tamil, and Tel. = Telugu

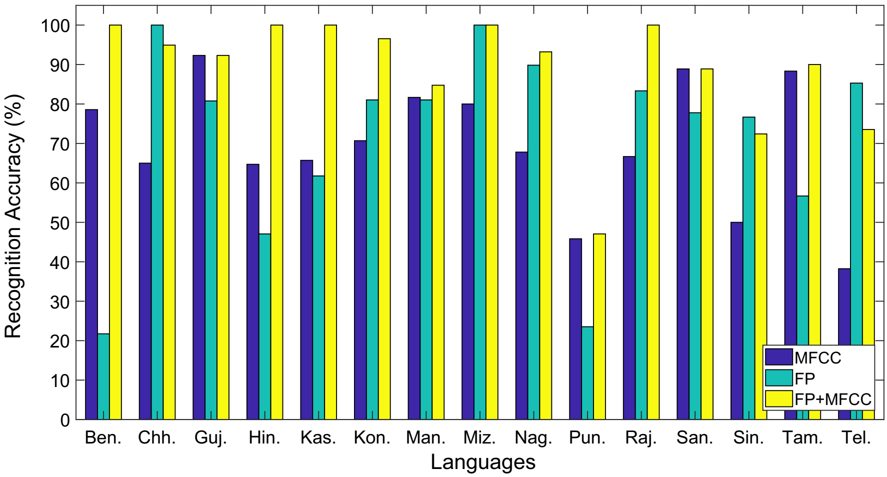

Fig. 13 Comparison of individual language recognition accuracies of best ANN classifiers with respect to global MFCC, FP, and MFCC + FP features on IITKGP-MLILSC database. The results depicted in this plot correspond to the models presented in Table 7, whose recognition accuracies are marked in bold. Ben. = Bengali, Chh. = Chhattisgarhi, Guj. = Gujarati, Hin. = Hindi, Kas. = Kashmiri, Kon. = Konkani, Man. = Manipuri, Miz. = Mizo, Nag. = Nagamese, Pun. = Punjabi, Raj. = Rajasthani, San. = Sanskrit, Sin. = Sindhi, Tam. = Tamil, and Tel. = Telugu

Table 9 Confusion matrix of ANN classifier for speaker-independent spoken language recognition using 120 FP features on AP18-OLR database (%)

	Can.	Ind.	Jap.	Kor.	Rus.	Vie.	Mand.	Kaz.	Tib.	Uyg.
Can.	**66.11**	3.06	9.67	2.00	1.06	1.20	1.67	5.56	2.50	7.17
Ind.	0.06	**98.94**	0.44	0.22		0.06	0.11			0.17
Jap.	5.33	4.56	**84.84**	1.89	0.07	0.14	0.93	1.05	0.14	1.05
Kor.	8.14	1.15	3.03	**79.39**	0.52	1.10	1.36	1.46	3.60	0.25
Rus.	4.02	0.05	4.07	2.14	**11.90**	0.21	14.40	10.86	12.58	39.77
Vie.	1.30	1.88	1.98		8.87	**81.59**	4.17	0.16		0.05
Mand.	5.78	6.45	2.45	14.95	7.34	10.06	**38.19**	3.61	7.17	4.00
Kaz.	2.40	1.39	1.45	0.22	1.73	0.84	14.11	**61.41**	5.47	10.98
Tib.	1.39	0.17		0.61	0.39		1.28	7.40	**87.20**	1.56
Uyg.	11.67	0.50	1.33	15.44	2.00	0.50	12.23	7.72	5.72	**42.89**

The rows and columns of the confusion matrix correspond to the targets and outputs, respectively. The diagonal elements are marked in bold, indicating the percentage of true classification with respect to individual languages. The off-diagonal elements indicate the percentage of mis-classification

Can. = Cantonese, Ind. = Indonesian, Jap. = Japanese, Kor. = Korean, Rus. = Russian, Vie. = Vietnamese, Mand. = Mandarin, Kaz. = Kazakh, Tib. = Tibetan, and Uyg. = Uyghur

database are depicted in a bar graph comparison plot as shown in Fig. 14. It is clear from Fig. 14 that the use of combined MFCC + FP features outperforms the use of MFCC and FP features, for most of the languages.

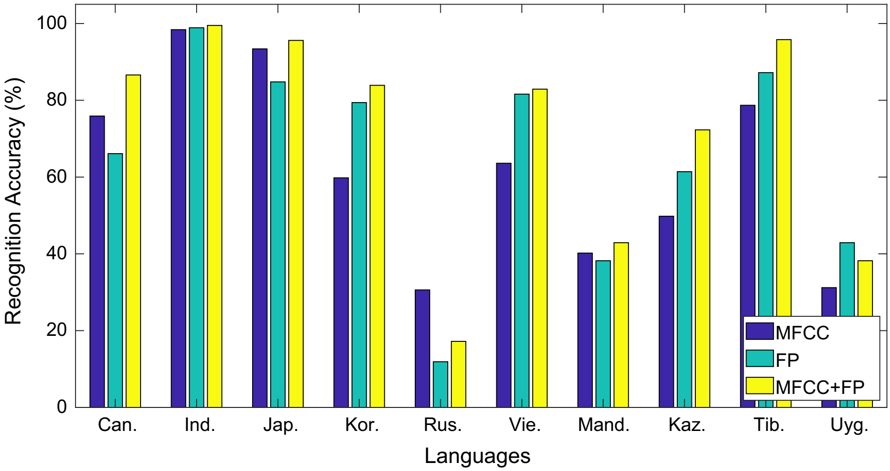

Fig. 14 Comparison of individual language recognition accuracies of best ANN classifiers with respect to global MFCC, FP, and MFCC + FP features on AP18-OLR database. The results depicted in this plot correspond to the models presented in Table 7, whose recognition accuracies are marked in bold. Can. = Cantonese, Ind. = Indonesian, Jap. = Japanese, Kor. = Korean, Rus. = Russian, Vie. = Vietnamese, Mand. = Mandarin, Kaz. = Kazakh, Tib. = Tibetan, and Uyg. = Uyghur

Table 10 Recognition performance of DNN (LSTM-RNN)-based spoken LID/LR systems

Database	Feature	Number of layers in LSTM-RNN & (number of LSTM and fully connected layers)					
		5 (1L-1FC)	6 (1L-2FC)	7 (1L-3FC)	6 (2L-1FC)	7 (2L-2FC)	8 (2L-3FC)
IITKGP-MLILSC	MFCC	**60.43**	59.22	55.05	58.95	49.39	54.37
	FP	60.70	58.82	55.59	58.55	**64.87**	58.14
	MFCC + FP	**84.52**	81.29	82.91	83.85	81.97	79.68
AP18-OLR	MFCC	54.63	**54.83**	56.67	54.61	54.12	53.79
	FP	59.05	**60.52**	58.94	59.41	58.24	60.38
	MFCC + FP	**68.54**	67.74	66.77	67.01	66.75	66.04

L. = LSTM layer, FC. = fully connected layer

Bold values indicate the maximum recognition accuracies of DNN (LSTM-RNN)-based spoken LID/LR systems with respect to databases and feature sets

6.3 DNN-Based Spoken LID/LR Systems

Table 10 presents the results of recognition accuracies achieved by LSTM-RNN classifiers using global MFCC, FP, and MFCC + FP features on IITKGP-MLILSC and AP18-OLR databases. The maximum recognition accuracies achieved with respect to different feature sets on both databases are marked in bold. As can be seen from Table 10, there is a nominal drop in the recognition performance with respect to an increase in the number of layers (either LSTM or FC or both), for most of the cases.

The use of LSTM-RNN networks with 5 layers (having 1 LSTM and 1 FC layer) and 6 layers (having 1 LSTM and 2 FC layers) is found to be sufficient for achieving reasonably good recognition performance. Comparatively, the networks with few layers are less complex and computationally efficient when compared to the networks with more layers.

In the case of IITKGP-MLILSC database, with respect to the proposed FP features, the LSTM-RNN network with 7 layers (having 2 LSTM and 2 FC layers) achieves the maximum recognition accuracy of 64.87%. The corresponding confusion matrix is shown in Table 11. As can be seen from Table 11, languages, namely Bengali, Hindi, Konkani, Punjabi, and Telugu, are majorly mis-classified with other languages. Similarly with respect to MFCC features, the LSTM-RNN network with 5 layers (having 1 LSTM and 1 FC layers) achieves the maximum recognition accuracy of 60.43%. The use of FP features shows an improvement in the recognition accuracy by 7.35% when compared to MFCC features. With respect to MFCC + FP features, the LSTM-RNN network with 5 layers (having 1 LSTM and 1 FC layers) achieves the maximum recognition accuracy of 84.52%. The use of MFCC + FP features shows a significant improvement in the recognition accuracies by 39.86% and 30.29% when compared to MFCC and FP features, respectively. Finally, the recognition accuracies achieved with respect to individual languages by the best LSTM-RNN networks trained with global MFCC, FP, and MFCC + FP features on IITKGP-MLILSC database are depicted in a bar graph comparison plot as shown in Fig. 15. It is clear from Fig. 15 that the use of combined MFCC + FP features outperforms the use of MFCC and FP features, for most of the languages.

In the case of AP18-OLR database, with respect to the proposed FP features, the LSTM-RNN network with 6 layers (having 1 LSTM and 2 FC layers) achieves the maximum recognition accuracy of 60.52%. The corresponding confusion matrix is shown in Table 12. As can be seen from Table 12, languages, namely Indonesian, Korean, Russian, and Mandarin, are majorly mis-classified with other languages. Similarly with respect to MFCC features, the LSTM-RNN network with 6 layers (having 1 LSTM and 2 FC layers) achieves the maximum recognition accuracy of 54.83%. The use of FP features shows an improvement in the recognition accuracy by 10.38% when compared to MFCC features. With respect to MFCC + FP features, the LSTM-RNN network with 5 layers (having 1 LSTM and 1 FC layers) achieves the maximum recognition accuracy of 68.54%. The use of MFCC + FP features shows a significant improvement in the recognition accuracies by 25.00% and 13.25% when compared to MFCC and FP features, respectively. Finally, the recognition accuracies achieved with respect to individual languages by the best LSTM-RNN networks trained with global MFCC, FP, and MFCC + FP features on AP18-OLR database are depicted in a bar graph comparison plot as shown in Fig. 16. It is clear from Fig. 16 that the use of combined MFCC + FP features outperforms the use of MFCC and FP features, for most of the languages.

6.4 Performance Comparison of Different Spoken LID/LR Models

Table 13 shows the comparison of recognition performance for the best spoken LID/LR systems in terms of databases, features, and classifiers. Table 13 summarizes the results

Table 11 Confusion matrix of DNN (LSTM-RNN) classifier for speaker-independent spoken language recognition using 120 FP features on IITKGP-MLILSC database (%)

	Ben.	Chh.	Guj.	Hin.	Kas.	Kon.	Man.	Miz.	Nag.	Pun.	Raj.	San.	Sin.	Tam.	Tel.
Ben.	**35.71**	2.38		28.57	21.43									11.91	
Chh.		**71.67**											25.00	3.33	
Guj.			**73.08**			1.92			9.62				15.38		
Hin.	38.24	14.71		**29.41**	2.93									14.71	
Kas.				35.29	**61.76**									2.95	
Kon.		1.72	48.28			**48.28**							1.72		
Man.		3.33			5.00		**58.33**	21.67					8.34	3.33	
Miz.							11.67	**80.00**					8.33		
Nag.			6.78						**93.22**						
Pun.					14.71		11.76			**29.41**			26.47		17.65
Raj.		1.67				1.67			3.33		**91.66**		1.67		
San.	27.78			11.11								**58.33**		2.78	
Sin.									11.67				**83.33**	5.00	
Tam.	20.00	1.67		13.33	1.67								1.67	**61.66**	
Tel.							2.94	41.18					8.82		**47.06**

The rows and columns of the confusion matrix correspond to the targets and outputs, respectively. The diagonal elements are marked in bold, indicating the percentage of true classification with respect to individual languages. The off-diagonal elements indicate the percentage of mis-classification

Ben. = Bengali, Chh. = Chhattisgarhi, Guj. = Gujarati, Hin. = Hindi, Kas. = Kashmiri, Kon. = Konkani, Man. = Manipuri, Miz. = Mizo, Nag. = Nagamese, Pun. = Punjabi, Raj. = Rajasthani, San. = Sanskrit, Sin. = Sindhi, Tam. = Tamil, and Tel. = Telugu

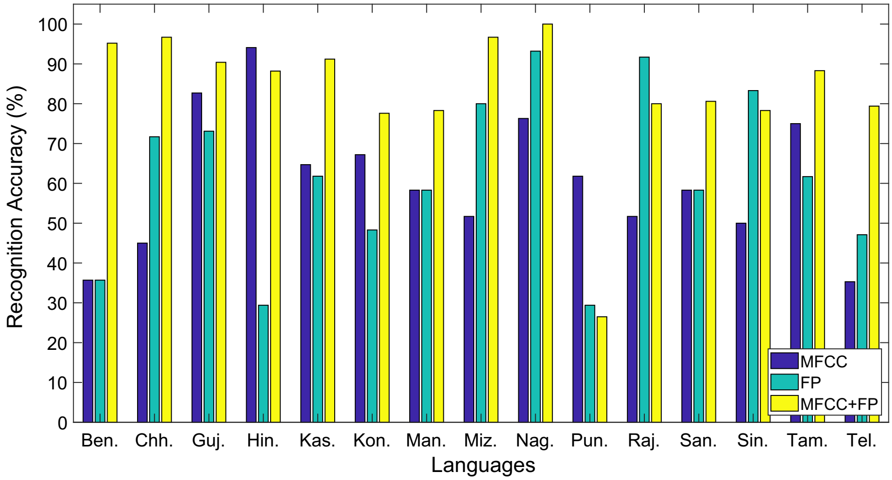

Fig. 15 Comparison of individual language recognition accuracies of best DNN (LSTM-RNN) classifiers with respect to global MFCC, FP, and MFCC + FP features on IITKGP-MLILSC database. The results depicted in this plot correspond to the models presented in Table 10, whose recognition accuracies are marked in bold. Ben. = Bengali, Chh. = Chhattisgarhi, Guj. = Gujarati, Hin. = Hindi, Kas. = Kashmiri, Kon. = Konkani, Man. = Manipuri, Miz. = Mizo, Nag. = Nagamese, Pun. = Punjabi, Raj. = Rajasthani, San. = Sanskrit, Sin. = Sindhi, Tam. = Tamil, and Tel. = Telugu

Table 12 Confusion matrix of DNN (LSTM-RNN) classifier for speaker-independent spoken language recognition using 120 FP features on AP18-OLR database (%)

	Can.	Ind.	Jap.	Kor.	Rus.	Vie.	Mand.	Kaz.	Tib.	Uyg.
Can.	**71.52**	0.94	0.41	0.89	6.16	4.69	0.73	8.87	1.88	3.91
Ind.	0.58	**4.60**	0.27	27.10	5.71	17.34	41.82	0.79	1.47	0.32
Jap.		9.18	**84.87**	1.51	0.74			2.14		1.56
Kor.	8.00	7.84	21.12	**23.85**	7.84	6.17	4.61	15.23	1.50	3.84
Rus.	1.78	1.39	0.17	28.28	**44.06**	4.96	9.82	7.08	0.73	1.73
Vie.	0.95	0.83	0.12	1.11	9.40	**81.91**	4.40	0.78	0.33	0.17
Mand.	5.33	1.17	0.06	16.11	17.44	4.78	**39.61**	1.06	0.94	13.50
Kaz.	0.83	1.28	2.11	2.28	1.95	0.33	4.56	**77.04**	1.50	8.12
Tib.	0.06	0.28	0.06	0.11	0.11		0.60	0.28	**97.11**	1.39
Uyg.	1.82	0.21	0.28	0.42	1.12	0.08	1.26	4.77	2.46	**87.58**

The rows and columns of the confusion matrix correspond to the targets and outputs, respectively. The diagonal elements are marked in bold, indicating the percentage of true classification with respect to individual languages. The off-diagonal elements indicate the percentage of mis-classification

Can. = Cantonese, Ind. = Indonesian, Jap. = Japanese, Kor. = Korean, Rus. = Russian, Vie. = Vietnamese, Mand. = Mandarin, Kaz. = Kazakh, Tib. = Tibetan, and Uyg. = Uyghur

presented in Tables 4, 7, and 10, by reporting the recognition accuracies of the best spoken LID/LR systems. The maximum recognition accuracies are marked in bold. As can be seen from Table 13, with respect to two databases (IITKGP-MLILSC and AP18-OLR) and three classifiers (SVM, ANN, and DNN), the use of FP features

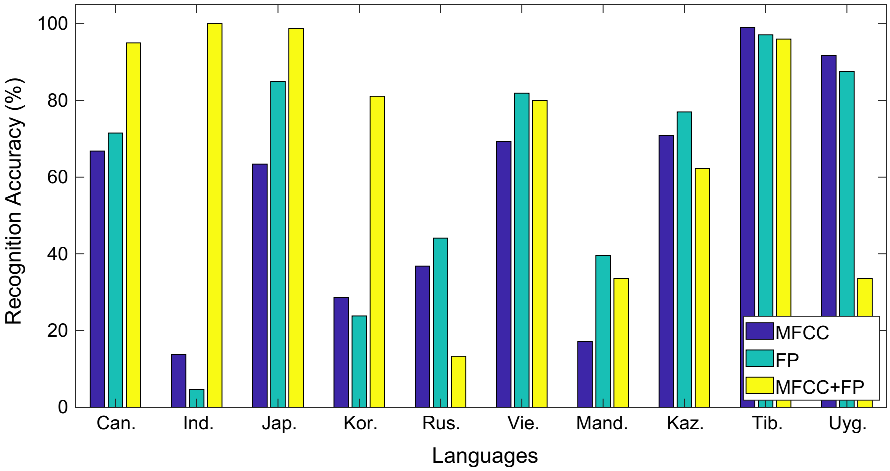

Fig. 16 Comparison of individual language recognition accuracies of best DNN (LSTM-RNN) classifiers with respect to global MFCC, FP, and MFCC + FP features on AP18-OLR database. The results depicted in this plot correspond to the models presented in Table 10, whose recognition accuracies are marked in bold. Can. = Cantonese, Ind. = Indonesian, Jap. = Japanese, Kor. = Korean, Rus. = Russian, Vie. = Vietnamese, Mand. = Mandarin, Kaz. = Kazakh, Tib. = Tibetan, and Uyg. = Uyghur

Table 13 Comparison of recognition performance achieved by the best spoken LID/LR systems in terms of databases, features, and classifiers

Database	Feature	SVM	ANN	DNN (LSTM-RNN)
IITKGP-MLILSC	MFCC	67.70	**71.10**	60.43
	FP	73.40	**74.20**	64.87
	MFCC + FP	86.40	**89.40**	84.52
AP18-OLR	MFCC	59.16	**61.30**	54.83
	FP	62.83	**64.70**	60.52
	MFCC + FP	70.73	**70.80**	68.54

Bold values indicate the maximum recognition accuracies of spoken LID/LR systems with respect to databases and feature sets

outperformed MFCC features, and the use of combined MFCC + FP features outperformed MFCC and FP features, for all the cases. In the case of IITKGP-MLILSC and AP18-OLR databases, the ANN-based spoken LID/LR systems trained with the combined MFCC + FP features achieve the maximum recognition accuracies of 89.40% and 70.80%, respectively.

Tables 14 and 15 show the optimal combination of features using MFCC, FP, and MFCC + FP, for different languages on IITKGP-MLILSC and AP18-OLR databases, respectively. In the case of SVM, ANN, and DNN classifiers, the use of MFCC + FP features shows a significant improvement in terms of individual language recognition accuracies when compared to MFCC and FP features. From the results presented in Tables 13, 14, and 15, it can be summarized that when compared to MFCC features, the FP features improved the recognition performance of the speaker-independent spoken

Table 14 Best features among MFCC, FP, and MFCC + FP for improved recognition with respect to individual languages of IITKGP-MLILSC database

Language	Feature		
	SVM	ANN	DNN
Bengali	MFCC + FP	MFCC + FP	MFCC + FP
Chhattisgarhi	FP	FP	MFCC + FP
Gujarati	MFCC/MFCC + FP	MFCC + FP	MFCC + FP
Hindi	MFCC + FP	MFCC + FP	MFCC
Kashmiri	MFCC + FP	MFCC + FP	MFCC + FP
Konkani	MFCC + FP	FP	MFCC + FP
Manipuri	MFCC + FP	MFCC	MFCC + FP
Mizo	FP/MFCC + FP	FP/FP + MFCC	MFCC + FP
Nagamese	MFCC + FP	FP	MFCC + FP
Punjabi	MFCC + FP	MFCC + FP	MFCC
Rajasthani	MFCC + FP	FP/MFCC + FP	FP
Sanskrit	MFCC/MFCC + FP	MFCC + FP	MFCC + FP
Sindhi	MFCC	MFCC + FP	FP
Tamil	MFCC + FP	MFCC + FP	MFCC + FP
Telugu	FP	MFCC + FP	MFCC + FP

Table 15 Best features among MFCC, FP, and MFCC + FP for improved recognition with respect to individual languages of AP18-OLR database

Language	Feature		
	SVM	ANN	DNN
Cantonese	FP	MFCC + FP	MFCC + FP
Indonesian	MFCC	MFCC + FP	MFCC + FP
Japanese	MFCC + FP	MFCC + FP	MFCC + FP
Korean	MFCC + FP	MFCC + FP	MFCC + FP
Russian	MFCC + FP	MFCC	FP
Vietnamese	MFCC + FP	MFCC + FP	FP
Mandarin	FP	MFCC + FP	FP
Kazakh	MFCC + FP	MFCC + FP	FP
Tibetan	MFCC + FP	MFCC + FP	MFCC
Uyghur	MFCC + FP	FP	MFCC

LID/LR systems. The recognition performance is further enhanced when MFCC and FP features are combined. The results presented in this paper with respect to IITKGP-MLILSC database can be compared with the existing results in the literature obtained using the same database. Table 16 compares the obtained results (corresponding to IITKGP-MLILSC database) of this paper with the results presented in [21,26,30].

It is clear from Table 16 that the proposed FP and the combination of MFCC + FP features improve the recognition performance of the spoken LID/LR systems when compared to the existing systems.

Table 16 Comparison of obtained results with few benchmark works in the literature using IITKGP-MLILSC database

Ref.	Description	Recog Perf. (%)
[21]	Speaker-independent LID/LR system (for 16 Indian languages)	36.85
	Features: MFCC. Classifier: GMM with 128 components	
	Speaker-independent LID/LR system (for 16 Indian languages)	40.68
	Features: LPCC. Classifier: GMM with 32 components	
	Speaker-independent LID/LR system (for 16 Indian languages) with proposed speaker-specific language models	44.92
	Features: MFCC. Classifier: GMM with 32 components	
	Speaker-independent LID/LR system (for 16 Indian languages) with proposed speaker-specific language models	35.96
	Features: LFCC. Classifier: GMM with 64 components	
	Speaker-independent LID/LR system (for 16 Indian languages) (k-best performance model, where $k = 3$)	61.84#
	Features: MFCC. Classifier: GMM with 32 components	
	Speaker-independent LID/LR system (for 16 Indian languages) (k-best performance model, where $k = 3$)	54.44#
	Features: LFCC. Classifier: GMM with 64 components	
[30]	Speaker-independent LID/LR system (for 27 Indian languages)	55.62*
	Features: MFCC derived from CBP. Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	58.65*
	Features: MFCC derived from pitch synchronous analysis (PSA). Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	61.06*
	Features: MFCC derived from glottal closure regions (GCR). Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	33.20*
	Features: intonation, rhythm, and stress features at syllable level. Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	34.65*
	Features: intonation, rhythm, and stress features at word level. Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	28.04*
	Features: prosodic features at phrase level using ΔF_0 contour. Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	24.77*
	Features: prosodic features at phrase level using duration contour. Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	21.08*
	Features: prosodic features at phrase level using ΔE contour. Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	33.26*
	Features: prosodic features at phrase level using ΔF_0+ duration $+\Delta E$ contour. Classifier: GMM	

Table 16 continued

Ref.	Description	Recog Perf. (%)
	Speaker-independent LID/LR system (for 27 Indian languages)	36.97*
	Features: combination of prosodic features at syllable + word levels. Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	38.94*
	Features: combination of prosodic features at syllable + word + phrase levels. Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	65.08*
	Features: combination of spectral and prosodic features. Classifier: GMM	
[26]	Speaker-independent LID/LR system (for 27 Indian languages)	19.00*
	Features: magnitude components of linear prediction (LP) residual signal represented by Hilbert envelope (HE), extracted at sub-segmental level. Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	39.33*
	Features: Magnitude components of LP residual signal represented by HE, extracted at segmental level. Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	29.33*
	Features: magnitude components of LP residual signal represented by HE, extracted at supra-segmental level. Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	49.33*
	Features: phase components of LP residual signal represented by residual phase (RP), extracted at sub-segmental level. Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	54.00*
	Features: phase components of LP residual signal represented by RP, extracted at segmental level. Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	50.00*
	Features: phase components of LP residual signal represented by RP, extracted at supra-segmental level. Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	56.67*
	Features: magnitude components of LP residual signal represented by HE, extracted at sub-segmental + segmental + supra-segmental levels. Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	69.67*
	Features: phase components of LP residual signal represented by RP, extracted at sub-segmental + segmental + supra-segmental levels. Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	69.67*
	Features: combined magnitude and phase components of LP residual signal represented by HE and RP, respectively, extracted at sub-segmental + segmental + supra-segmental levels. Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages)	51.00*

Table 16 continued

Ref.	Description	Recog Perf. (%)
	Features: combined magnitude and phase components of LP residual signal represented by HE and RP, respectively, extracted at sub-segmental level. Classifier: GMM	
	Speaker-independent LID/LR system (for 27 Indian languages) Features: combined magnitude and phase components of LP residual signal represented by HE and RP, respectively, extracted at segmental level. Classifier: GMM	55.33*
	Speaker-independent LID/LR system (for 27 Indian languages) Features: combined magnitude and phase components of LP residual signal represented by HE and RP, respectively, extracted at supra-segmental level. Classifier: GMM	52.00*
	Speaker-independent LID/LR system (for 27 Indian languages) Features: combined evidences of composite magnitude and phase components of LP residual signal represented by HE and RP, respectively, extracted at sub-segmental + segmental + supra-segmental levels. Classifier: GMM	70.33*
In this Paper	Speaker-independent LID/LR system (for 15 Indian languages) Features: Global MFCC. Classifier: DNN (5-Layer LSTM-RNN), having 1 LSTM and 1 FC layer	60.43
	Speaker-independent LID/LR system (for 15 Indian languages) Features: global MFCC. Classifier: SVM with OVO configuration	67.70
	Speaker-independent LID/LR system (for 15 Indian languages) Features: global MFCC. Classifier: ANN with 2 and 3 hidden layers	71.10
	Speaker-independent LID/LR system (for 15 Indian languages) Features: global FP. Classifier: DNN (7-Layer LSTM-RNN), having 2 LSTM and 2 FC layer	64.87
	Speaker-independent LID/LR system (for 15 Indian languages) Features: global FP. Classifier: SVM with OVA configuration	73.40
	Speaker-independent LID/LR system (for 15 Indian languages) Features: global FP. Classifier: ANN with 1 hidden layer	74.20
	Speaker-independent LID/LR system (for 15 Indian languages) Features: global MFCC + FP. Classifier: DNN (5-Layer LSTM-RNN), having 1 LSTM and 1 FC layer	84.52
	Speaker-independent LID/LR system (for 15 Indian languages) Features: global MFCC + FP. Classifier: SVM with OVA configuration	86.40
	Speaker-independent LID/LR system (for 15 Indian languages) Features: global MFCC + FP. Classifier: ANN with 1 hidden layer	89.40

Ref. = Reference. Recog Per. = Recognition performance (%)

#The recognition performances are quoted with respect to seven Indian languages (Bengali, Gujarati, Hindi, Kashmiri, Punjabi, Tamil, and Telugu) used in this paper

*The recognition performances are quoted with respect to 15 Indian languages (Bengali, Chhattisgarhi, Gujarati, Hindi, Kashmiri, Konkani, Manipuri, Mizo, Nagamese, Punjabi, Rajasthani, Sanskrit, Sindhi, Tamil, and Telugu) used in this paper

The recognition performances presented in this table are quoted with respect to the test speech signals of 10-s duration

This concludes that the FP features are efficient in terms of discriminating languages and can be employed effectively to develop robust spoken LID/LR systems.

7 Conclusion and Future Work

In previous studies, different frame-based spectral features have been used for spoken language recognition. In this paper, a new Fourier parameter (FP) model is proposed to extract salient features from acoustic speech signals, for capturing the language-specific information. The proposed features are evaluated on two multilingual speech databases, namely IITKGP-MLILSC and AP18-OLR databases, in Indian and oriental languages, respectively.

Harmonic amplitude FPs are estimated from every frame of the speech signal. Initially, the characteristics of the harmonic amplitude FPs are studied by considering the mean statistical parameter. At first, one particular harmonic amplitude FPs are extracted from the frames of speech signals corresponding to a single speaker from each language of both databases. Their corresponding means are computed across speech signals with respect to frames. It is observed that the amplitudes vary with respect to different languages. The similar kind of variation is observed in the case of other harmonics. An adequate number of harmonic amplitudes FPs are necessary since it is difficult to classify/recognize signals based on the features obtained from single harmonics. So initially, the first 120 harmonic amplitude FPs are extracted from a randomly chosen speech signal of each language for both databases. Their corresponding means are computed across frames. Interesting characteristics are observed from the maximum peaks of the mean harmonic amplitudes. The maximum peaks of the mean harmonic amplitudes corresponding to the speech signals of different languages which occur at the same or adjacent harmonics are grouped together in the form of clusters. The clusters so formed contain majority of the peaks corresponding to a distinct linguistic family. This phenomenon is observed in a number of trails performed with a randomly chosen speech signal from each language of both databases. The distinct characteristics exhibited by FPs show that there is a relationship associated with FPs and the language traits. This relationship is exploited to develop robust spoken LID/LR models.

The global features are known to provide superior performance. Therefore, the statistical parameters like the mean, median, standard deviation, minimum, and maximum of 120 FP features across frames are computed to derive global FP features. The computed global FP features, along with their associated first-order and second-order differences, are used to construct a 1800-dimensional global FP feature vector. MFCC features are also extracted alongside with FP features for performance comparison. The 39-dimensional MFCC features are extracted and later transformed to a 195-dimensional global MFCC features by considering the same statistical parameters (used in the case of global FP features). Later, the ReliefF feature selection algorithm is used to reduce the dimension of global MFCC and FP features by ignoring irrelevant, noisy, and redundant features. From the estimates of ReliefF feature selection, top 100 and 900 discriminative features (having higher weights and lower ranks) are selected from 195 global MFCC and 1800 global FP features. The global MFCC and

FP feature sets with reduced dimensions are used to develop spoken LID/LR models in Indian and oriental languages for 15 and 10 languages, respectively.

Spoken LID/LR models are developed using SVM-, ANN-, and DNN (LSTM-RNN)-based classifiers. The models are independently trained and tested using global MFCC, FP, and MFCC + FP features. In the case of IITKGP-MLILSC database, the experimental results show that the proposed FP features improve the recognition accuracies over MFCC features by 8.42%, 4.36%, and 7.35% (with respect to SVM, ANN, and DNN classifiers). The use of combined MFCC + FP features improves the recognition accuracies over MFCC features by 27.62%, 25.74%, and 39.86% (with respect to SVM, ANN, and DNN classifiers) and over FP features by 17.71%, 20.49%, and 30.29% (with respect to SVM, ANN, and DNN classifiers).

Similarly, in the case of AP18-OLR database, the experimental results show that the proposed FP features improve the recognition accuracies over MFCC features by 6.20%, 5.55%, and 10.38% (with respect to SVM, ANN, and DNN classifiers). The use of combined MFCC + FP features improves the recognition accuracies over MFCC features by 19.56%, 15.50%, and 25.00% (with respect to SVM, ANN, and DNN classifiers), and over FP features by 12.57%, 9.43%, and 13.25% (with respect to SVM, ANN, and DNN classifiers).

The experimental results show that the proposed FP features are very much effective in characterizing and recognizing languages from speech signals when compared to MFCC features. Moreover, the FP features also assist in improving the performance of language recognition when they are combined with MFCC features. The obtained results establish that the proposed FP features are useful for spoken language recognition.

This paper analyzes the performance of combining MFCC and FP features to develop robust spoken LID/LR systems in Indian and oriental languages. The present study can be extended to develop and analyze the performance of different spoken LID/LR systems using the combination of FP and other sophisticated language-specific spectral and prosodic features with different classification models. One can investigate the robustness and degree of contribution made by each type of feature sets in improving the recognition performance of the systems. Development of robust LID/LR models can be explored for noisy environments.

Acknowledgements The authors would like to thank all anonymous reviewers for providing their valuable comments on earlier drafts of this manuscript. Their comments and suggestions are very useful and helped us to improve the quality of the final manuscript. The authors express out appreciation to Prof. Dr. K. Sreenivasa Rao and his research team for sharing IITKGP-MLILSC database with us during the course of this research. The authors also express out appreciation to Dr. Zhiyuan Tang for sharing AP18-OLR (AP16-OL7 and AP17-OL3) multilingual database with us during the course of this research. The authors would also like to thank MathWorks®, Inc., for providing MATLAB® tool and NCH®, Inc., for providing WavePad® Sound Editor Tool. Any correspondence should be made to N. S. Sai Srinivas.

References

1. F. Adeeba, S. Hussain, Acoustic feature analysis and discriminative modeling for language identification of closely related South-Asian languages. Circuits Syst. Signal Process. **37**(8), 3589–3604 (2018). https://doi.org/10.1007/s00034-017-0724-1

2. E. Ambikairajah, H. Li, L. Wang, B. Yin, V. Sethu, Language identification: a tutorial. IEEE Circuits Syst. Mag. **11**(2), 82–108 (2011). https://doi.org/10.1109/MCAS.2011.941081
3. J.C. Ang, A. Mirzal, H. Haron, H.N.A. Hamed, Supervised, unsupervised, and semi-supervised feature selection: a review on gene selection. IEEE/ACM Trans. Comput. Biol. Bioinform. **13**(5), 971–989 (2016). https://doi.org/10.1109/TCBB.2015.2478454
4. M.E. Ayadi, M.S. Kamel, F. Karray, Survey on speech emotion recognition: features, classification schemes, and databases. Pattern Recognit. **44**(3), 572–587 (2011). https://doi.org/10.1016/j.patcog.2010.09.020
5. J. Balleda, H.A. Murthy, Language identification from short segment of speech, in *Proc. ICSLP-2000*, vol. 3, pp. 1033–1036 (2000)
6. J. Benesty, M.M. Sondhi, Y. Huang, *Springer Handbook of Speech Processing* (Springer, Berlin, 2008). https://doi.org/10.1007/978-3-540-49127-9
7. C.C. Bhanja, M.A. Laskar, R.H. Laskar, A pre-classification-based language identification for North-east Indian languages using prosody and spectral features. Circuits Syst. Signal Process. (2018). https://doi.org/10.1007/s00034-018-0962-x
8. C. Busso, S. Mariooryad, A. Metallinou, S. Narayanan, Iterative feature normalization scheme for automatic emotion detection from speech. IEEE Trans. Affect. Comput. **4**(4), 386–397 (2013). https://doi.org/10.1109/T-AFFC.2013.26
9. C. Cortes, V. Vapnik, Support-vector network. Mach. Learn. **20**, 273–297 (1995). https://doi.org/10.1007/BF00994018
10. S. Davis, P. Mermelstein, Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE Trans. Acoust. Speech Signal Process. **28**(4), 357–366 (1980). https://doi.org/10.1109/TASSP.1980.1163420
11. A. Geron, *Hands-on Machine Learning with Scikit-Learn and TensorFlow* (O'Reilly Media, Newton, 2017)
12. J. Gonzalez-Dominguez, I. Lopez-Moreno, H. Sak, J. Gonzalez-Rodriguez, P.J. Moreno, Automatic language identification using long short-term memory recurrent neural networks, in *Interspeech 2014, 15th Annual Conference of the International Speech Communication Association, Singapore*, pp. 2155–2159 (2014). https://www.isca-speech.org/archive/interspeech_2014/i14_2155.html
13. M.T. Hagan, H.B. Demuth, M.H. Beale, O.D. Jesus, *Neural Network Design*, 2nd edn. (Martin Hagan, Boston, 2014)
14. C.W. Hsu, C.J. Lin, A comparison of methods for multiclass support vector machines. IEEE Trans. Neural Netw. **13**(2), 415–425 (2002). https://doi.org/10.1109/72.991427
15. S. Jothilakshmi, V. Ramalingam, S. Palanivel, A hierarchical language identification system for Indian languages. Digit. Signal Process. **22**(3), 544–553 (2012). https://doi.org/10.1016/j.dsp.2011.11.008
16. V. Kecman, T.M. Huang, M. Vogt, *Iterative Single Data Algorithm for Training Kernel Machines from Huge Data Sets: Theory and Performance* (Springer, Berlin, 2005), pp. 255–274. https://doi.org/10.1007/10984697_12
17. D.P. Kingma, J.L. Ba, ADAM: A method for stochastic optimization. Computing Research Repository (CoRR) abs/1412.6980, arXiv:1412.6980 (2014)
18. S.G. Koolagudi, A. Bharadwaj, Y.V.S. Murthy, N. Reddy, P. Rao, Dravidian language classification from speech signal using spectral and prosodic features. Int. J. Speech Technol. **20**(4), 1005–1016 (2017). https://doi.org/10.1007/s10772-017-9466-5
19. M. Leena, K.S. Rao, B. Yegnanarayana, Neural network classifiers for language identification using phonotactic and prosodic features, in *Proceedings of 2005 International Conference on Intelligent Sensing and Information Processing*, pp. 404–408 (2005). https://doi.org/10.1109/ICISIP.2005.1529486
20. H. Li, B. Ma, K.A. Lee, Spoken language recognition: from fundamentals to practice. Proc. IEEE **101**(5), 1136–1159 (2013). https://doi.org/10.1109/JPROC.2012.2237151
21. S. Maity, A.K. Vuppala, K.S. Rao, D. Nandi, IITKGP-MLILSC speech database for language identification, in *2012 National Conference on Communications (NCC)*, pp. 1–5 (2012). https://doi.org/10.1109/NCC.2012.6176831, https://ieeexplore.ieee.org/document/6176831/
22. K.E. Manjunath, K.S. Rao, Improvement of phone recognition accuracy using articulatory features. Circuits Syst. Signal Process. **37**(2), 704–728 (2018). https://doi.org/10.1007/s00034-017-0568-8
23. M.F. Møller, A scaled conjugate gradient algorithm for fast supervised learning. Neural Netw. **6**(4), 525–533 (1993). https://doi.org/10.1016/S0893-6080(05)80056-5

24. K.V. Mounika, A. Sivanand, H.R. Lakshmi, V.G. Suryakanth, V.A. Kumar, An investigation of deep neural network architectures for language recognition in indian languages, in *Interspeech 2016*, pp. 2930–2933 (2016). https://doi.org/10.21437/Interspeech.2016-910
25. T. Nagarajan, H.A. Murthy, Language identification using spectral vector distribution across languages, in *Proceedings of International Conference on Natural Language Processing* (2002)
26. D. Nandi, D. Pati, K.S. Rao, Language identification using Hilbert envelope and phase information of linear prediction residual, in *2013 International Conference Oriental COCOSDA Held Jointly with 2013 Conference on Asian Spoken Language Research and Evaluation (O-COCOSDA/CASLRE)*, pp. 1–6 (2013). https://doi.org/10.1109/ICSDA.2013.6709864, https://ieeexplore.ieee.org/document/6709864
27. A.V. Oppenheim, R.W. Schafer, *Discrete-Time Signal Processing* (Prentice Hall, Upper Saddle River, NJ, 1999)
28. K.S. Rao, Application of prosody models for developing speech systems in Indian languages. Int. J. Speech Technol. **14**(1), 19–33 (2011). https://doi.org/10.1007/s10772-010-9086-9
29. K.S. Rao, S. Sarkar, *Robust Speaker Recognition in Noisy Environments* (Springer, Berlin, 2014). https://doi.org/10.1007/978-3-319-07130-5
30. V.R. Reddy, S. Maity, K.S. Rao, Identification of Indian languages using multi-level spectral and prosodic features. Int. J. Speech Technol. **16**(4), 489–511 (2013). https://doi.org/10.1007/s10772-013-9198-0
31. F. Richardson, D. Reynolds, N. Dehak, Deep neural network approaches to speaker and language recognition. IEEE Signal Process. Lett. **22**(10), 1671–1675 (2015). https://doi.org/10.1109/LSP.2015.2420092
32. M. Robnik-Šikonja, I. Kononenko, Theoretical and empirical analysis of ReliefF and RReliefF. Mach. Learn. **53**(1), 23–69 (2003). https://doi.org/10.1023/A:1025667309714
33. K.G. Sheela, S.N. Deepa, Review on methods to fix number of hidden neurons in neural networks. Math. Probl. Eng. **2013**(425740), 1–11 (2013). https://doi.org/10.1155/2013/425740
34. M. Siu, X. Yang, H. Gish, Discriminatively trained GMMs for language classification using boosting methods. IEEE Trans. Audio Speech Lang. Process. **17**(1), 187–197 (2009). https://doi.org/10.1109/TASL.2008.2006653
35. Sreevani, C.A. Murthy, Bridging feature selection and extraction: Compound feature generation. IEEE Trans. Knowl. Data Eng. **29**(4), 757–770 (2017). https://doi.org/10.1109/TKDE.2016.2619712
36. N.S.S. Srinivas, N. Sugan, L.S. Kumar, M.K. Nath, A. Kanhe, Speaker-independent Japanese isolated speech word recognition using TDRC features, in *2018 International CET Conference on Control, Communication, and Computing (IC4)*, pp. 278–283 (2018). https://doi.org/10.1109/CETIC4.2018.8530947, https://ieeexplore.ieee.org/document/8530947
37. N. Sugan, N.S.S. Srinivas, N. Kar, L.S. Kumar, M.K. Nath, A. Kanhe, Performance comparison of different cepstral features for speech emotion recognition, in *2018 International CET Conference on Control, Communication, and Computing (IC4)*, pp. 266–271 (2018). https://doi.org/10.1109/CETIC4.2018.8531065, https://ieeexplore.ieee.org/document/8531065
38. Z. Tang, D. Wang, Y. Chen, Q. Chen, AP17-OLR challenge: Data, plan, and baseline. in *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pp. 749–753 (2017). https://doi.org/10.1109/APSIPA.2017.8282134, https://ieeexplore.ieee.org/document/8282134
39. Z. Tang, D. Wang, Q. Chen, AP18-OLR challenge: three tasks and their baselines, in *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pp. 596–600 (2018). https://doi.org/10.23919/APSIPA.2018.8659714
40. V.N. Vapnik, *Statistical Learning Theory* (Wiley, New York, 2001)
41. M.K. Veera, R.K. Vuddagiri, S.V. Gangashetty, A.K. Vuppala, Combining evidences from excitation source and vocal tract system features for Indian language identification using deep neural networks. Int. J. Speech Technol. **21**(3), 501–508 (2018). https://doi.org/10.1007/s10772-017-9481-6
42. R.K. Vuddagiri, K. Gurugubelli, P. Jain, H.K. Vydana, A.K. Vuppala, IIITH-ILSC speech database for Indian language identification, in *The 6th International Workshop on Spoken Language Technologies for Under-Resourced Languages*, pp. 56–60 (2018)
43. D. Wang, L. Li, D. Tang, Q. Chen, AP16-OL7: a multilingual database for oriental languages and a language recognition baseline, in *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pp. 1–5 (2016). https://doi.org/10.1109/APSIPA.2016.7820796, https://ieeexplore.ieee.org/document/7820796

44. K. Wang, N. An, B.N. Li, Y. Zhang, L. Li, Speech emotion recognition using Fourier parameters. IEEE Trans. Affect. Comput. **6**(1), 69–75 (2015). https://doi.org/10.1109/TAFFC.2015.2392101

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Affiliations

N. S. Sai Srinivas[1] · N. Sugan[1] · Niladri Kar[1] · L. S. Kumar[1] · Malaya Kumar Nath[1] · Aniruddha Kanhe[1]

✉ N. S. Sai Srinivas
satya_srinivasnettimi@live.com
http://sites.google.com/site/nssaisrinivas/

N. Sugan
sugannece@gmail.com

Niladri Kar
niladri1357@gmail.com

L. S. Kumar
lakshmi@nitpy.ac.in

Malaya Kumar Nath
malaya.nath@nitpy.ac.in

Aniruddha Kanhe
aniruddhakanhe@nitpy.ac.in

[1] Department of Electronics and Communication Engineering, National Institute of Technology Puducherry Karaikal, Thiruvettakudy, Karaikal, Union Territory of Puducherry 609 609, India