A GMM supervector approach for spoken Indian language identification for mismatch utterance length

Aarti Bakshi[1], Sunil Kumar Kopparapu[2]
[1]Department of Electronics and Communication Engineering, UMIT, SNDT University, Mumbai, India
[2]TCS Innovations Laboratories, Thane, India

Article Info

Article history:

Received Sep 30, 2020
Revised Dec 15, 2020
Accepted Jan 19, 2021

Keywords:

Artificial neural network
GMM-UBM
GMM-UBM supervectors
Spoken language identification
Support vector machine

ABSTRACT

Gaussian mixture model-universal background model (GMM UBM) supervectors are used to identify spoken Indian languages. The supervectors are calculated from short-time MFCC, its first and sec derivatives. The UBM builds a generalized Indian language model, and mean adaptation transforms it to a duration normalized language-specific GMM. Multi-class support vector machine and artificial neural network classifiers are used to identify language labels from the supervectors. Experimental evaluations are performed using 30 sec speech utterances from nine Indian languages comprised five Indo-Aryan and four Dravidian languages, extracted from all India radio broadcast news data-set. Eight smaller duration data-sets were manually derived to study the effect of training and test duration mismatch. In mismatch conditions, identification accuracy decreases with a decrease in test and train utterance duration. Investigations showed that the 32-mixture model with ANN classifier has optimal performance.

Corresponding Author:

Aarti Bakshi
Department of Electronics and Communication Engineering
SNDT University Santacruz (w), Mumbai, Maharashtra, India
Email: aarti.bakshi@kccemsr.edu.in

1. INTRODUCTION

The spoken language identification (SLID) system recognizes the language, from the desired set, by analyzing a short-duration spoken utterance. An SLID system enables the automatic selection of language and grammar models to convert speech to text in conversational interfaces like siri, alexa, and google home. In a vernacular call center, a SLID system can be used to route the incoming call to a human agent conversantly communicating in the customer's native language. Spoken languages vary in dialects and accents, which poses challenges in building an efficient SLID system [1]. In India, a multilingual country, most of the official languages can be grouped into two families, Indo-Aryan and Dravidian. SLID systems based on Indian languages are motivated because the languages belonging to different families (inter-family) are relatively easier to identify than the languages belonging to the same family (intra-family).

A SLID system can be explained in two phases: (i) the training phase and (ii) the testing phase. A language identification model is trained in the training phase by extracting language-specific features from the speech utterance. In the testing phase, the trained model's performance is evaluated using utterances that agnostic to training. Several features reported in the SLID system literature can be broadly grouped into two categories: (a) low-level speech features and (b) high-level speech features. The low-level features exploit the phonetic nature of Indian languages. It consists of phono-acoustic, phototactic, and prosodic features. Phono-acoustic features compare the frequency of occurrence of fundamental phonemes to distinguish languages. Phone recognition and its parallel version, followed by language modeling, are most widely used for the phonotactic approach. Although SLID systems based on phonetically transcribed speech utterances are accurate, the data is not readily available [2]. Such systems are also prone to errors in manual transcription and phone recognition. Prosodic features discriminate languages based on long term characteristics like tone [3], rhythm [4], duration [5], energy, and pitch contour. The use of speech production model-based features like linear cepstral coefficients (LPCC) [6], perceptual linear prediction (PLP) [7], and Fourier features [8] have been reported in the literature. The efficiency in native recognition problems inspired the use of perception based mel-frequency cepstral coefficients (MFCC) with Δ and Δ^2 for SLID tasks [9, 10]. The importance of temporal information, suggested by MFCC and its derivatives, motivated the use of shifted delta cepstral coefficients (SDC) [11] in SLID systems. It was reported that the performance of MFCC based systems decreases with decreasing frame size [9, 10].

Classifiers like Hidden Markov Model (HMM) [12], vector quantization (VQ) [6], support vector machine (SVM) [3, 13, 14], artificial neural network (ANN) [15, 16], and Gaussian mixture model (GMM) [15-17] have been reported to model feature vectors in SLID systems. One of the simplest techniques used for the SLID system is GMM-UBM. In this method, maximum likelihood estimation is used to train the language model, and maximum a posterior (MAP) estimation is used to adapt the UBM model. The speech sample is a series of the independent spectral feature vector, and GMM mathematically models these features with UBM adaption known as GMM-UBM supervectors carries spectral characteristics [10-18]. These features are adapted to UBM using the MAP estimation algorithm to obtain utterance-based GMM [19]. GMM-UBM supervector performs well on short utterance length and decides it by calculating the likelihood ratio using spectral features. A comparison of under complete dictionary problem using GMM mean shifted supervector and overcomplete i-vector approach for spare classification were addressed in the [20]. In this approach, GMM mean shifted supervector was obtained using a concatenation of the mean vector of the mean of adapted GMM-UBM, which shows superior performance over the i-vector approach. Bhattacharyya-based GMM system was developed using an adaptive relevance factor to address negative effects on the language characteristics. The author tried to address duration variability for the individual utterance of 30 and 10 sec [21].

In practical SLID systems, such as a vernacular call center, may fulfill the requirement of training speech utterance duration, but equally long test speech utterance may not be available. It has been reported that the SLID system's performance degrades with the increasing mismatch between durations of training and test speech utterances [22]. The paper presents GMM-UBM based SLID system for nine Indian languages under matched and mismatched training and test utterance duration. SLID systems trained with long segment length utterances are known to perform well, but it will become worst when short segment length utterance is counted [22]. We conducted a series of experiments for different utterance length mismatch cases on eight different segment length utterance data-sets to analyze this. It is observed that with a sufficient amount of training data, the GMM-UBM supervector performs very well for short segment length utterance. The rest of the paper is structured as follows. Section 2 discussed the proposed SLID system using Indian languages; section 3 describes the experimental setup and results using ANN and OvA SVM. We conclude in section 4.

2. PROPOSED SLID SYSTEM

The architecture of SLID using the Indian language is shown in Figure 1. The first step in the process is to develop a data-set for nine different languages, Assamese (A_S), Bengali (B_N), Gujarati (G_J), Hindi (H_N), Marathi (M_R), Kannada (K_N), Malayalam (M_L), Tamil (T_M), and Telugu (T_L). It split into a training data-set and testing data-set using a 5-fold cross-validation process. The second step involved feature extraction, which converts speech waveform into parametric representation. Each spoken utterance is processed using framing and windowing functions. Mel frequency cepstral coefficients (MFCC) features are computed from each frame and append with delta and acceleration (Δ and Δ^2) coefficients. A total of 39 MFCC feature vectors from all 9 Indian languages are used to develop the GMM-UBM model. The language-specific GMM model is developed by adapting trained UBM using MAP method. Note that the supervector maps an utterance to a high-dimensional vector. We adapt the mean of Gaussian components for each speech utterance using the MAP algorithm and concatenating mean vectors of all Gaussian components formed GMM-UBM supervectors. It forms ($39 \times M$) GMM-UBM supervector matrix per language. ANNs and OvA multi-class SVM are trained to predict the class (language).

Extracting the original signal's meaningful characteristics, thereby representing the original signal with a lesser amount of data without any major loss in the original signal's information, are referred to as feature extraction. Universal background model (UBM) is typically used to model the data distribution and is very popular in speaker recognition. GMM is used to capture characteristics of the language-independent features. For a *K*-dimensional language-specific feature vector x_k, the Gaussian mixture density is represented as (1) [3].

$$P(x_k \,|\lambda) = \sum_{i=1}^{K} w_i b_i x_k \tag{1}$$

where $x_{k,}$ k=1, 2, …, K is K dimensional feature vectors and $b_i x_k$, where i=1, 2…K is component densities and w_i , where i=1, 2,…K is the mixture weights, respectively.
The mixture weights (w_i), mean vectors (μ_i) characterize GMM, and covariance matrices ($\sum_i$) is represented as (2).

$$\lambda = w_i \mu_i \textstyle\sum_i \tag{2}$$

The maximum likelihood estimation algorithm aims to estimate λ (language model), which maximizes the likelihood of GMM for the set of training data. In this work, X represents the acoustic vectors obtained from MFCC features to compute the GMM likelihood is represented as (3).

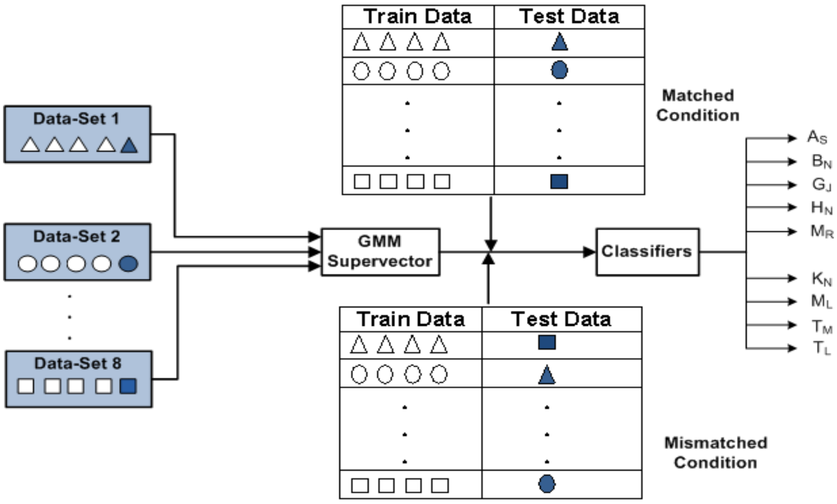

Figure 1. The architecture of the proposed SLID system using Indian languages

$$P(\mathrm{X}|\lambda) = \prod_{k=1}^{T}(x_k \,|\lambda) \tag{3}$$

In this paper, maximum likelihood estimation is calculated using an iterative expectation maximum (EM) method. The basic concept behind the EM method is, to begin with, basic model λ and estimate a new model λ` such that $(P(X|\lambda) < P(X|\overset{\prime}{\lambda}))$. This model will become a basic model for the next iteration, and error between basic model parameters and the new model start reducing, and this process will repeat until a definite threshold value is achieved. For a given spoken utterance s and estimated language L, the language identification system's role is to determine *s* belongs to L or not. For a given feature space X, GMM models the feature vector of spoken utterance for estimated H_0 such that λ_L is the estimated language corresponding to spoken utterance s. Another estimated H_1 in the same feature space is represented as the likelihood ratio is defined as (4).

$$Ir\ (\mathrm{X}) = \frac{p(\mathrm{X}|\lambda_L)}{p(\mathrm{X}|\lambda'_L)} \tag{4}$$

The GMM is trained using EM method to compute the language model parameter λ using MAP approach [3, 13]. To compute the statistics of GMM-UBM mixture components, the probabilistic alignment of the training vectors needs to be calculated as weight (w_i), means (E_i) and variance (E_i^2).

$$w_i = \sum_{k=1}^{T} p_r\,(i|x_k) \tag{5}$$

$$E_i = \frac{1}{w_i}\sum_{k=1}^{T} p_r\,(i, x_k)x_k \tag{6}$$

$$E_i^2 = \frac{1}{w_i}\sum_{k=1}^{T} p_r\,(i, x_k)x_k^2 \tag{7}$$

In the case of the SLID system, maximum likelihood (ML) algorithm is used to determine the model's parameters developed and MAP algorithm is used to derive the model using UBM adaption by calculating means μ_i of GMM [3].

The biological neural network influences an ANN with three layers, namely, input, output, and hidden. Each layer is made up of several neurons. Typically, the feature vector's length determines the number of neurons at the input layer, and the number of classes to identify decides the number of the neurons' output layers. The experimental analysis decides the number of neurons and hidden layers. ANN is trained using a backpropagation algorithm which works on the principle of gradient descent. The backpropagation algorithm picks the error and is fed back to the network to modify the network's weights, which will ensure a small loss in the next iteration. This process is repeated iteratively, and updation of the weights ensures a better match between the expected and the network output [3, 23].

SVM is generally used for binary classification. It performs non-linear classification with a kernel trick that maps the feature vectors to a high dimensional feature space. Multi-class SVM is designed by combining several binary classifiers and usually is extended to handle multiple classes [13]. These methods are proven to be expensive than the binary classification problem but show faster convergence in handling the same amount of data. A *d*-class One-vs-All (OvA) SVM constructs *d* binary classifiers. Each binary SVM classifier is trained the *i*th class training data labeled as positive and all other *(d-1)* classes labeled as negative. The *i*th class test data can be identified by the binary classifier with positive *i*th class labeled as positive [24]. The most commonly used kernel function in SVM is linear, polynomial, and Gaussian kernels that map the low dimension feature vectors to high dimensional feature vectors. The use of SVM for language identification has two advantages; first, it can be used to solve the multi-class problem, and the second one can handle a sequence of feature vectors.

3. RESULTS AND DISCUSSION

All experimental evaluations are carried out using own speech corpus developed using all India radio audio files. It comprised 900 audio recordings of news bulletins, each of 30 sec duration and sampling frequency of 16 kHz, read by male and female newsreaders in nine Indian languages [25]. The language selection was based on their phoneme sound distribution, and they belong to language families being spoken by a large population [1] . The languages can be grouped as: Indo-Aryan family and Dravidian family. The Indo-Aryan family consists of Assamese (A_S), Bengali (B_N), Gujarati (G_J), Hindi (H_N), and Marathi (M_R). The Dravidian family consists of Kannada (K_N), Malayalam (M_L), Tamil (T_M), and Telugu (T_L). Each utterance of 30 sec was manually split into utterances of smaller length to derive seven new speech corpuses of 0.2 sec, 0.5 sec, 1 sec, 3 sec, 5 sec, 10 sec, and 15 sec. Each utterance was manually inspected and utterances containing music, unwanted voices, and long silences were removed.

A 5-fold cross-validation was used to avoid overfitting and measuring the SLID system's accuracy independent of training-test split. ANN (regularization value: 0.1, activation function: ReLu, number of epochs: 200) and multi-class SVM (with Gaussian kernel, OvA decomposition, regularization factor: 1.3) are classification models were trained with the feature vectors as the input and the corresponding language label (one of nine languages) as output.

a. Match condition

Initially data-set was divided into three sets: developments, training, and testing sets with 50%, 30%, and 20% data. The number of mixtures M in GMM-UBM was varied as 8, 16, 32, and 64. Table 1 shows the performance evaluation of eight data-sets when training and testing data have the same duration i.e. matched condition. The optimum number of neurons and hidden layers required was found out experimentally.

The performance of SLID system increased with increasing number of mixtures with maximum accuracy of 99.9% at 64 mixtures for 30 sec data-set using ANN classifier. The lowest accuracy of 36% at 32 mixtures for 0.2 sec data-set used OvA SVM classifier. The experimental evaluation explored the use of GMM-UBM supervector approach with ANN and OvA SVM models to solve the problem of short test utterances. A slight increment in the accuracy was observed with increase in the length of utterances. As expected more reliable system can be develop with long length utterances. Table 2 shows a comparison of the accuracy of GMM-UBM supervector based ANN with earlier approaches in the literature for 30 sec data-set. Table 3 compares accuracy GMM-UBM supervector based ANN with earlier approaches in the literature shows that GMM-UBM supervector based ANN with short length utterances 0.2 sec and 0.5 sec an accuracy of 76.1% and 90.2% is achieved respectively. Figure 2 shows ROC curve of 0.2 sec data-set using ANN (green line) and OvA SVM (brown line) for nine Indian languages. The performance of the ANN is marginally better than OvA SVM.

Table 1. Accuracy (%) of GMM-UBM supervector based SLID system using ANN and OvA SVM for Indian languages under matched condition

Data-set	ANN				OvA SVM			
(sec)	M=8	M=16	M=32	M=64	M=8	M=16	M=32	M=64
30	99.1	99.5	99.3	99.9	99.7	99.7	99.2	99.7
15	98.2	99.3	99.3	99.6	99.4	99.7	99.7	99.6
10	98.0	98.9	99.1	99.4	99.6	99.3	99.2	99.5
5	98.7	98.2	98.2	99.1	98.2	98.3	95.5	96.5
3	95.5	97.6	98.1	98.0	99.3	96.4	95.1	95.8
1	92.6	95.1	95.7	96.0	92.9	94.2	94.7	94.7
0.5	84.9	89.2	90.1	90.2	75.3	72.0	75.8	73.9
0.2	69.4	70.4	75.5	76.1	41.2	39.4	36.0	37.0

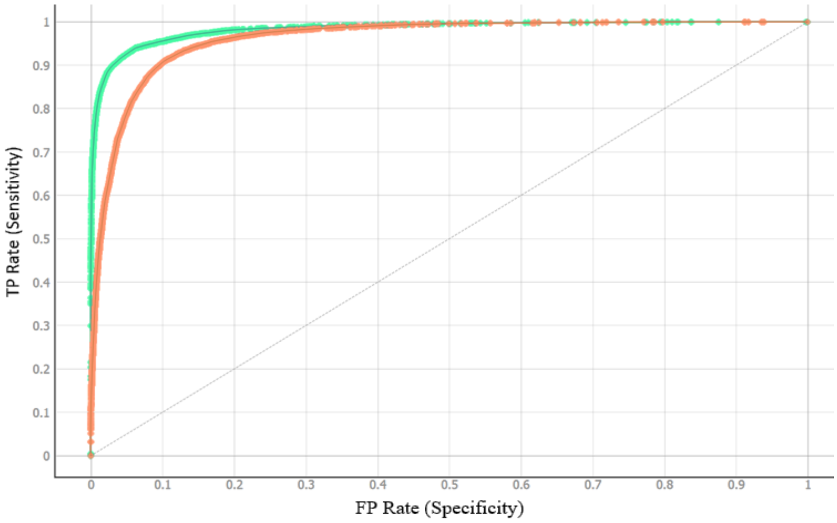

Figure 2. ROC curve of 0.2 sec data-set using ANN (green line) and OvA SVM (brown line)

Table 2. Comparison of proposed GMM-UBM supervector based ANN with other approaches in literature

Approach	Accuracy (%)
GMM supervector based ANN	99.9
BNF based HDAE [26]	97.1
i-vector based DNN [15]	90.8
MFCC-SDC based GMM-UBM [19]	76.35
MFCC-SDC with i-vector [19]	50.45

Table 3. Comparison of proposed GMM-UBM supervector based ANN with other approaches in literature for short utterances

Utterance length (sec)	Approach	Accuracy (%)
0.2	GMM supervector based ANN	76.1
0.5	GMM supervector based ANN	90.2
0.4	GFCC + MFCC based BLSTM [27]	50.0
0.5	MFCC + SDC based LSTM-RNN [28]	50.0

b. Mismatch condition

This section investigates the performance of the SLID model in different segment length utterance train-test conditions. Table 4 supports the observation for 30 sec segment length utterance duration training-

0.2, 0.5, 1, 3, 5, 10, 15 sec segment length utterance duration testing condition. Here 4 folds (80% spoken utterances) of 30 sec segment length data-set were used to train the classifier and remaining 1 fold (20% spoken utterances) of the data-set was used for testing. In mismatch train-test segment length utterances, test utterances of different segments length obtained by splitting utterances in testing fold of 30 sec data-set were used.

Table 4 depicts that relative improvement in the recognition accuracy with Gaussian mixtures decreases with the reduction in the segment length utterances. The best performance is achieved using OvA SVM for 15 sec and 0.5 sec training data-set while it drastically degraded for 0.2 sec data-set. The results show the system's encouraging performance for short utterance length for test conditions when system is trained with long utterance length.

Table 4. Accuracy (%) of GMM-UBM supervector based SLID system using ANN and OvA SVM for Indian languages for 30 sec segment length utterance used for training

Data-set (sec)	ANN				OvA SVM			
	M=8	M=16	M=32	M=64	M=8	M=16	M=32	M=64
15	84.7	89.6	95.6	95.6	91.9	83.4	94.5	96.7
10	84.6	90.5	91.9	92.7	89.6	92.9	93.8	95.6
5	80.5	84.9	84.9	86.1	82.9	85.6	84.5	90.5
3	67.7	63.6	61.1	75.0	74.9	74.5	66.6	77.5
1	49.6	49.3	44.2	46.6	41.0	44.1	40.1	43.5
0.5	43.6	44.5	41.5	43.2	47.2	48.7	45.4	48.1
0.2	23.2	23.7	24.4	24.7	22.2	23.1	22.4	23.9

The experimental results reported in Tables 5 and 6 show a comparison of segment length utterances mismatch condition. Each row of the table indicates the segment length utterance used to train the classifiers, and SLID recognition accuracy columns indicate how accurately our model could classify the correct language. We expect to have high recognition accuracy on diagonal (match condition) with respect to off-diagonal (mismatch condition). The system performance degrades when trained with 30, 15, 10, 5, 3, and 1sec and tested with 0.2 sec. This is because 0.2 sec segment length utterance carries less language discriminative information. In very short utterances, especially (≤ 3 sec), GMM-UBM supervector with ANN works better than multi-class SVM.

Table 5. SLID system performance for mismatch segment length utterance condition using ANN (%)

Training dataset	Test dataset (sec)							
(sec)	30	15	10	5	3	1	0.5	0.2
30	98.0	84.7	84.6	75.0	67.6	49.6	43.6	23.2
15	96.3	96.5	91.1	91.6	65.0	51.2	49.5	30.1
10	86.7	96.2	96.3	91.9	63.0	50.8	52.3	35.7
5	95.6	98.9	98.9	93.1	74.8	61.7	55.1	38.2
3	91.1	88.5	89.0	84.7	92.0	62.2	63.8	40.3
1	84.7	94.9	95.3	95.9	73.6	91.3	83.2	46.2
0.5	84.7	94.9	95.3	95.3	82.9	93.6	81.7	50.1
0.2	47.3	44.8	45.2	45.9	45.2	53.4	61.9	64.0

Table 6. SLID system performance for mismatch segment length utterance condition using OvA SVM (%)

Training dataset	Test dataset (sec)							
(sec)	30	15	10	5	3	1	0.5	0.2
30	81.6	89.6	91.6	82.9	74.9	41.0	47.2	22.2
15	94.9	84.6	98.5	95.0	73.9	48.1	49.8	23.1
10	84.4	97.8	78.5	95.9	74.0	48.3	51.4	25.2
5	82.2	98.6	98.9	74.0	81.4	62.2	53.2	28.4
3	87.8	85.5	90.1	90.8	72.2	64.4	55.3	30.2
1	81.8	72.1	77.3	89.7	76.6	61.6	58.2	32.2
0.5	80.1	70.2	75.2	86.3	73.3	59.8	60.0	33.6
0.2	20.1	22.2	24.3	26.4	28.3	30.8	33.2	39.6

Overall results show some specific observations. The results of all tables' show that GMM-UBM supervector based ANN and OvA SVM worked significantly better on the long segment length utterances for both match and mismatch conditions. In a real-time application, designing the SLID system works on short segment length utterances is more desirable. However, the system performance degrades for short segment length utterances of 0.5 and 0.2 sec used to train and test the classifiers under utterance length mismatch condition. For the utterance length match condition, GMM-UBM based ANN worked better than GMM-UBM based OvA SVM.

4. CONCLUSION

A short-time MFCC and its first and second derivative based GMM-UBM supervectors for SLID system using Indian languages have been presented. GMM-UBM supervector with ANN and multi-class SVM classifiers were compared for matched training-test duration and mismatched training-test duration. In matched conditions, the performance of the GMM-UBM supervector with ANN was similar to multi-class SVM for long segment length utterances; however, for short segment length utterances, ANN performs better than multi-class SVM. In mismatched conditions, GMM-UBM supervector with ANN performed better than multi-class SVM; however, it degrades when the test segment length utterance is below 3 sec. The effect of very short duration utterances on system performance needs to be further investigated. Other feature extraction techniques in the GMM-UBM framework need to be explored. The results show that the SLID system using Indian languages will have promising applications in vernacular call centers, speech recognition.

REFERENCES

[1] B. Aarti and S. K. Kopparapu, “Spoken Indian language identification: a review of features and databases,” *Sādhanā*, vol. 43, no. 4, 2018.

[2] E. Ambikairajah, *et al*., “Language Identification: A Tutorial,” *IEEE Circuits and Systems Magazine*, vol. 11, no. 2, pp. 82-108, 2011.

[3] A. Bakshi and S. K. Kopparapu, “Spoken Indian Language Classification using GMM supervectors and Artificial Neural Networks,” in *2019 IEEE Bombay Section Signature Conference (IBSSC)*, 2019, pp. 1-6.

[4] J. L. Rouas and J. Farinas, “Automatic Modelling of Rhythm and Intonation for Language Identification,” in *15th International Congress of Phonetic Sciences (15th ICPhS)*, 2003, pp. 567-570.

[5] V. R. Reddy, S. Maity, and K. S. Rao, “Identification of Indian languages using multi-level spectral and prosodic features,” *International Journal of Speech Technology*, vol. 16, no. 4, pp. 489-511, 2013.

[6] E. Mansour, *et al*., “LPC and MFCC Performance Evaluation with Artificial Neural Network for Spoken Language Identification,” *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 6, no. 3, 2013.

[7] R. Cole, *et al*., “Language identification with neural networks: a feasibility study,” in *Conference Proceeding IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, 1989, pp. 525-529.

[8] N. S. S. Srinivas, *et al*., “Recognition of Spoken Languages from Acoustic Speech Signals Using Fourier Parameters,” *Circuits, Systems, and Signal Processing*, vol. 38, no. 11, pp. 5018-5067, 2019.

[9] S. G. Koolagudi, D. Rastogi, and K. S. Rao, “Spoken Language Identification Using Spectral Features,” *Communications in Computer and Information Science Contemporary Computing*, pp. 496-497, 2012.

[10] R. Tong, *et al*., “Integrating Acoustic, Prosodic and Phonotactic Features for Spoken Language Identification,” in *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, Toulouse, 2006, pp. I-I.

[11] M. Itrat, *et al*., “Automatic Language Identification for Languages of Pakistan,” *International Journal of Computer Science and Network Security (IJCSNS)*, vol. 17, no. 2, pp. 161, 2017.

[12] R. K. Vuddagiri, H. K. Vydana, and A. K. Vuppala, “Improved Language Identification Using Stacked SDC Features and Residual Neural Network,” *The 6th Intl. Workshop on Spoken Language Technologies for Under-Resourced Languages*, pp. 210-214, 2018.

[13] E. Noor and H. Aronowitz, “Efficient Language Identification using Anchor Models and Support Vector Machines,” in *2006 IEEE Odyssey - The Speaker and Language Recognition Workshop*, San Juan, 2006, pp. 1-6.

[14] D. Sengupta and G. Saha, “Automatic recognition of major language families in India,” in *2012 4th International Conference on Intelligent Human Computer Interaction (IHCI)*, Kharagpur, 2012, pp. 1-4.

[15] C. Bhanja, *et al*., “A Pre-classification-Based Language Identification for Northeast Indian Languages Using Prosody and Spectral Features,” *Circuits, Systems, and Signal Processing*, vol. 38, no. 5, pp. 2266-2296, 2019.

[16] S. Jothilakshmi, V. Ramalingam, and S. Palanivel, “A hierarchical language identification system for Indian languages,” *Digital Signal Processing*, vol. 22, no. 3, pp. 544-553, 2012.

[17] V. R. Kumar, H. K. Vydana, and A. K. Vuppala, “Significance of GMM-UBM based Modelling for Indian Language Identification,” *Procedia Computer Science*, vol. 54, pp. 231-236, 2015.

[18] P. A. Torres-Carrasquillo, *et al*., “Approaches to language identification using Gaussian mixture models and shifted delta cepstral features,” *INTERSPEECH*, 2002.

[19] F. Adeeba and S. Hussain, “Acoustic Feature Analysis and Discriminative Modeling for Language Identification of Closely Related South-Asian Languages,” *Circuits, Systems, and Signal Processing*, vol. 37, no. 8, pp. 3589-3604, 2018.

[20] O. P. Singh, B. C. Haris, and R. Sinha, “Language identification using sparse representation: A comparison between GMM supervector and i-vector based approaches,” in *2013 Annual IEEE India Conference (INDICON)*, Mumbai, 2013, pp. 1-4, doi: 10.1109/INDCON.2013.6726125.
[21] C. H. You, H. Li, and K. A. Lee, “A GMM-supervector approach to language recognition with adaptive relevance factor,” in *2010 18th European Signal Processing Conference*, Aalborg, 2010, pp. 1993-1997.
[22] A. Poddar, M. Sahidullah, and G. Saha, “Performance comparison of speaker recognition systems in presence of duration variability,” in *2015 Annual IEEE India Conference (INDICON)*, New Delhi, 2015, pp. 1-6, doi: 10.1109/INDICON.2015.7443464.
[23] B. Aarti and S. K. Kopparapu, “Spoken Indian Language Classification Using ANN and Multi-Class SVM,” in *2018 International Conference On Advances in Communication and Computing Technology (ICACCT)*, Sangamner, 2018, pp. 213-218.
[24] J. Salomon, “Support Vector Machines for Phoneme Classification,” Master thesis, Master of Science School of Artificial Intelligence, Division of Informatics, University of Edinburgh, 2003.
[25] A. Bakshi and S. K. Kopparapu, “Spoken Indian Language Identification”, June 19, 2020, IEEE Dataport, doi: https://dx.doi.org/10.21227/xm4q-s210.
[26] H. S. Das and P. Roy, “Bottleneck Feature-Based Hybrid Deep Autoencoder Approach for Indian Language Identification,” *Arabian Journal for Science and Engineering*, vol. 45, no. 4, pp. 3425-3436, 2020.
[27] F. Adeeba and S. Hussain, “Native Language Identification in Very Short Utterances Using Bidirectional Long Short-Term Memory Network,” *IEEE Access*, vol. 7, pp. 17098-17110.
[28] R. Zazo, *et al*., “Language Identification in Short Utterances Using Long Short-Term Memory (LSTM) Recurrent Neural Networks,” *Plos One*, vol. 11, no. 1, 2016.

BIOGRAPHIES OF AUTHORS

Aarti Bakshi is currently pursuing a Ph.D. from UMIT, SNDT University, Mumbai. She has completed her BE (Electronics Engineering) from Pune University and ME (Electronics and Telecommunication) from the University of Mumbai. Her current area of interest includes speech processing, language recognition, image processing, and machine learning. She has several papers in international conferences, journals. She is a Life member of ISTE and IETE.

Sunil K. Kopparpu is a Principal Scientist with TCS Innovations Laboratories, Mumbai. He received his Ph. D degree from IIT Bombay. He has several conferences, journals, publications, and patents. He is co-author of the book at Springer brief. His area of interest is the image, speech, and natural language processing. His is a member of IEEE.