IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 14, No. 2, April 2025, pp. 1561∼1567
ISSN: 2252-8938, DOI: 10.11591/ijai.v14.i2.pp1561-1567

Hindi spoken digit analysis for native and non-native speakers

Parabattina Bhagath[1], Malempati Shanmukha[1], Pradip K. Das[2]
[1]Department of Computer Science and Engineering, Lakireddy Bali Reddy College of Engineering, Andhra Pradesh, India
[2]Department of Computer Science and Engineering, Indian Institute of Technology Guwahati, Assam, India

Article Info

Article history:

Received Mar 15, 2024
Revised Nov 11, 2024
Accepted Nov 24, 2024

Keywords:

Convolutional neural networks
Digit recognition
Hindi speech
Mel frequency cepstral coefficients
Under-resourced speech recognition

ABSTRACT

Automated speech recognition (ASR) is the process of using an algorithm or automated system to recognize and translate spoken words of a specific language. ASR has various applications in fields such as mobile speech recognition, the internet of things and human-machine interaction. Researchers have been working on issues related to ASR for more than 60 years. One of the many use cases of ASR is designing applications such as digit recognition that aid differently-abled individuals, children and elderly people. However, there is a lack of spoken language data in under-developed and low-resourced languages, which presents difficulties. Although this is not a pivotal issue for highly established languages like English, it has a significant impact on less commonly spoken languages. In this paper, we discuss the development of a Hindi-spoken dataset and benchmark spoken digit models using convolutional neural networks (CNNs). The dataset includes both native and non-native Hindi speakers. The models built using CNN exhibit 88.44%, 95.15%, and 89.41% for non-native, native, and combined speakers respectively.

Corresponding Author:

Parabattina Bhagath
Department of Computer Science and Engineering, Lakireddy Bali Reddy College of Engineering
Mylavaram, Andhra Pradesh, India
Email: bhagath.2014@alumni.iitg.ac.in

1. INTRODUCTION

Spoken data sets are essential for conducting research in the field of speech processing. This research involves studying spoken units and developing new features and modeling techniques to address various problems such as recognition, verification, emotion recognition, and smart interfaces for internet of things (IoT) applications. To obtain reasonable knowledge, a substantial amount of data is required, including spoken datasets like vowels, words, digits, numbers, and sentences. Each application requires a distinct dataset for designing speech models and open-source datasets are preferable as they are freely available to the research community. However, this requirement is still unmet for low-resource and under-resourced languages, especially Asian languages like Japanese, Chinese, and various Indian languages. India, with its diverse cultures and languages, has a unique dialect for each language. Nevertheless, a majority of these languages do not contribute much to automated spoken language understanding, despite having rich grammar and vocabulary [1].

Mobile speech recognition has developed into an intriguing field that enables users to access several technical features. Speech processing can assist in giving extra features to portable devices, especially for people who are physically impaired. The implementation method must be extremely tailored to the language in order to support the creation of applications. As a result, IoT devices must support natively spoken languages in order to accommodate ignorant consumers' demands. Native spoken languages must be employed with these technologies in order to make the applications easier for end users to use [2]. Numbers are very important entities in various applications that includes dialling phone numbers and PIN numbers. Therefore, the digit recognition problem has a potential impact. However, the problem is marginally addressed in the literature on Indian languages. In this paper, we developed a Hindi digit dataset and investigated various activation functions to find the suitable convolutional neural network (CNN) architecture for developing speaker-independent digit recognition models.

The paper is structured as follows: section 2 describes the previous work on the speech corpus available in the literature. In section 3, the precise process followed in the work is elaborated. Section 4 discusses the experimental environment and the results obtained. Finally, section 5 concludes the paper.

2. RELATED WORK

Over the years, hidden Markov models (HMMs) have been widely successful in modeling speech across different languages. These models have been utilized in the development of speech recognition and speaker verification systems for Indian languages. However, recent research suggests that deep learning models are also effective in addressing speech-processing challenges. Tailor *et al.* [3] introduced a deep learning model for recognizing spoken digits in the Gujarati language, utilizing CNN in combination with mel frequency cepstral coefficients (MFCC) as speech features. The study reported an accuracy of 98.7% using a dataset of 2,400 samples [3]. Another survey on Gujarati was carried out by Dalsaniya *et al.* [4], during which they created a Gujarati digit database with recordings from 20 speakers.

An isolated digit recognition system that uses multiple features is proposed for the Bundelkhandi language in [5]. A modified MFCC algorithm is proposed by adding a 1-D median filter in this work. Works on spoken digit recognition have been found in the Odia language. The authors employed CNN and MFCC for modeling and feature extraction respectively. The results are encouraging with a recognition rate of 91.72% for a small dataset with 500 samples [6]. Bengali is the language spoken in the regions of West Bengal, Assam and a few northeastern states of India and Bangladesh. Sharmin *et al.* [7] proposed a deep learning-based recognition system for Bengali digits where authors have observed a 98.37% recognition rate. Another work on Bangla accent is reported by the authors where HMMs and MFCCs were used for modeling and feature extraction. A recognition rate of 90% was observed with the approach [8].

Telugu is a widely spoken language in the region of Andhra Pradesh, Telangana and Puducherry. There are a few works reported on Telugu digit classification. Bhagath *et al.* [9] developed a Telugu spoken digit corpus [9] consisting of around 50 speakers and a benchmark model was proposed in [10] that uses HMMs. In another work of the same authors, a CNN-based recognition model was proposed. The recognition rate was observed to be 80% and 73% for the HMM and CNN respectively [11]. A phoneme segmentation algorithm for the Indian-accented English digits was proposed in [12]. The segmentation algorithm finds the boundaries between the phonetic units in a word.

Kumar and Aggarwal [13] proposed a recurrent neural network (RNN)-based recognition system for Hindi speech recognition. The approach uses wavelets as the prominent features and is integrated with MFCC and gammatone frequency cepstral features (GFCC) as the prominent features along with the HMM. In this work, the dataset collected from 100 speakers was used [13]. Sangeetha *et al.* [14] addressed the problem of connected digits of Hindi and Tamil languages using long short-term memory (LSTM) networks and the system achieved 72% accuracy for Hindi digits. Saxena and Wahi [15] developed a Hindi recognition system using the HTK Toolkit to build speaker-dependent models. The model proposed by the authors used MFCCs as acoustic features. The models exhibited an 85% recognition rate on a very limited dataset consisting of 5 speakers [15]. Hussain and Roy [16] proposed a neuro-fuzzy approach for spoken digit recognition for the Hindi language with a dataset of 4000 samples. The approach was able to recognize the digits in 99% of the cases [16].

Aside from the Indian languages, the problem of digit recognition has also been a challenge in many other languages worldwide. Wazir and Chuah [17] proposed an LSTM approach for Arabic digit classification. Their dataset comprised of 1040 spoken Arabic utterances of the digits. The study found that the approach was successful in 80% of the cases [17]. Other works on Arabic digits can be found in [18], [19]. In addition to Arabic, research on the Chinese language is also ongoing [20]. CNNs were originally developed for image-processing tasks, but they have also been applied to various speech-processing problems. The effectiveness of CNNs for speech recognition depends on three main attributes: pooling, weight sharing, and locality.

The pooling layer in a CNN enables the network to handle frequency shifts commonly present in speech signals. Since speech recognition models need to learn from multiple frequencies, weight-sharing mechanisms can help address this requirement. Additionally, the locality property helps reduce the number of weights necessary for the network [21]. Researchers have introduced various CNN models suitable for speech processing problems. Li and Zhou [22] proposed a command recognition approach using CNNs, which utilize MFCCs as the primary features for representing spoken commands. Pereira et al. examined the robustness for keyword spotting in embedded speech recognition systems using multiple frameworks such as Lenet-5, SqueezeNet, and EfficientNet-B0. They discovered that SqueezeNet was an efficient model for this task [23]. Wubet *et al.* [24] introduced a CNN-LSTM combined model for accent classification, designed for non-native English speakers.

Recognizing digits is an essential aspect of speech processing. However, no recent prominent research has been found on the Hindi language. Hindi-spoken digit recognition is a well-known problem in the field of speech recognition. In this paper, we propose a speaker-independent digit recognition model designed using the Hindi speech data spoken by native and non-native speakers from different parts of India. The next section describes the detailed methodology of the proposed approach.

3. METHODOLOGY

3.1. Data set preparation

The dataset collection procedure consists of several stages, beginning with training and culminating in an essential stage called labelling. Initially, a group of speakers was selected from various regions of India. Following that, the spoken utterances were recorded. The spoken data is then validated for accuracy during the validation stage. Finally, the data is correctly labelled, resulting in the creation of the final dataset. This process is summarized in Figure 1. The current dataset contains both native and non-native speakers of the Hindi language from various states in the Indian subcontinent, including Andhra Pradesh, Telangana, Delhi, Bihar, Assam, and Madhya Pradesh. The speakers are graduate students from different states in India, between the ages of 19 and 22, and include both females and males. The recording sessions conducted to collect the dataset are shown in Figure 2. The format for labelling the data is shown in Figure 3.

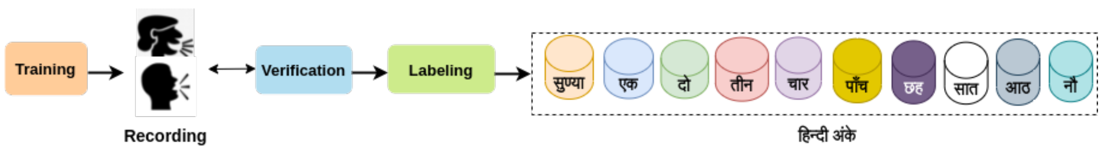

Figure 1. Flowchart of the dataset collection

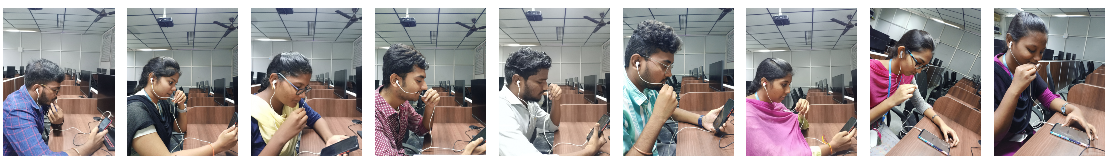

Figure 2. Recording sessions

Speaker ID	Word Label	Utterance Number

Figure 3. Labeling the spoken digits

The speaker's individual identification number, Speaker ID is the first element. The utterance number of the word is the second component. One of the following strings will be present in this part of the label: {'0', '1', '2', '3', '4', '5', '6', '7', '8', '9'}. Finally, the utterance number will start with '00' and end with '19'. For example, the first recording for the word "Zero" of the speaker with id "19761A05T9" is named "19761A0568_0_00.wav". In total, 29,000 voice samples of 10 different digits of 145 different speakers make up the data collection. The summary of the dataset is given in Table 1. The data set is made available at the Hindi data set: https://tinyurl.com/2p8xjwc8.

Table 1. Dataset summary

S. No	Attributes	Values
1	Number of digits in the dataset	10
2	Distinct speakers	145
3	Sessions held	80
4	Number of utterances	29000
5	Duration of the dataset in hours	16.1

3.2. Feature extraction

The second step in the approach is to extract the essential features from the raw speech signals. The present approach considers the MFCC feature extraction procedure. Broadly, the shape of the power spectrum is captured for acquiring the patterns residing in a speech signal. The computation of the required features starts by first pre-processing the raw speech signal. This step normalizes the speech signals and brings them to a uniform range of amplitudes. Once the pre-processing is complete, each signal is divided into short segments of 20-30ms windows. After segmenting, a hamming window represented by (1) is applied to remove the effects of abrupt changes at the signal edges. This follows extracting the frequency components and it is done by computing the discrete fourier transform (DFT) of the source signal. For any speech signal $x[n]$, the DFT can be calculated using the (2).

$$h[x] = (1 - \delta) - \delta cos\left(\frac{2\pi n}{L-1}\right) \tag{1}$$

$$X[k] = \sum_{n=0}^{N-1} e^{\frac{-2\pi jn}{N}} \tag{2}$$

The third step involves computing the power spectrum of the DFT. Finally, the mel-scale is applied to the power spectrum. This step ensures that the frequency components obtained from the speech signals fall within the range of human hearing perception. The mel-scale of $x[n]$ can be represented as given in (3). The process of MFCC extraction is depicted in Figure 4. Each speech segment gives 13 MFCC features and the features for all words are arranged in a matrix form with the class labels. This will be subsequently used in modeling as a feature matrix. The complete procedure for modeling is explained in the next subsection.

$$M(x[n]) = 1127ln(1 + x[n]/700) \tag{3}$$

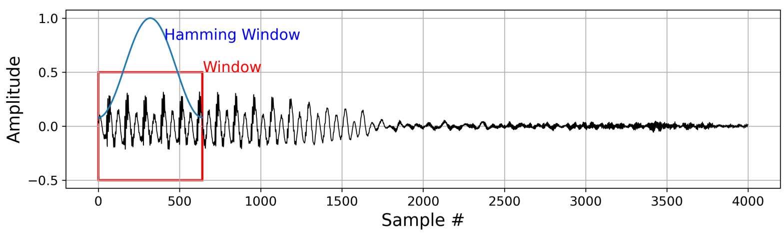

Figure 4. Frame-wise MFCC feature extraction from a source speech signal

3.3. Digit modeling with convolutional neural network

The model building starts by obtaining the features using the procedure discussed in section 3.2.. The proposed CNN architecture for modelling Hindi digits is shown in Figure 5. The method will process the feature matrix through a series of layers called hidden layers. Each hidden layer has convolution, pooling and activation functions. The proposed digit recognition architecture has 3 such layers as shown in Figure 5. First, the input feature matrix is convoluted with a kernel represented as a square matrix. The convolution is a simple cross-correlation between two matrices. This operation is followed by a pooling step in which the resultant matrix obtained from the convolution operation is down-sampled using statistics such as max, min, and average. The results obtained from the pooling operation are filtered through a non-linear function called the activation function. Generally, the classification capability of a CNN depends on the type of activation functions used. The selection of the activation function is a crucial step in designing the CNN architecture. In the present work, we studied the behaviour of the network with different combinations of activation functions [25] at different layers and the one with the highest recognition rate was chosen. The activation functions used for the study are shown in Table 2. Each function exhibits different nature and the performance of the model varies with the activation function being used at different layers. The results observed with the activation functions are discussed in the next section.

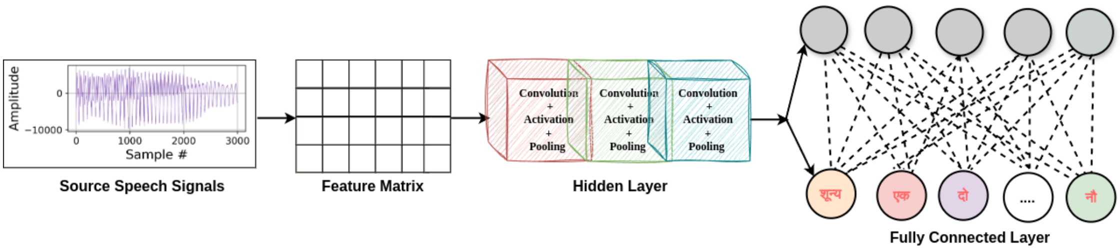

Figure 5. CNN model for digit recognition

Table 2. Activation functions used in different models

S. No	Activation function	Description
1.	Softplus	$softplus(X_i) = log(1 + e^{X_i})$
2.	Softmax	$softmax(X_i) = \frac{e^{X_i}}{\sum_{j=1}^{N} e^{X_j}}$
3.	tanh	$tanh(X_i) = \frac{e_i^{x} - e^{-x_i}}{e_i^{x} + e^{-x_i}}$
4.	RELU	$RELU(X_i) = max(0, X_i)$
5.	ELU	$ELU(X_i) = x_i$ if $X_i > 0, \alpha e^{X_i} - 1$ if $X_i \leq 0$
6.	SELU	$SELU(X_i) = \lambda X_i$ if $X_i \geq 0, SELU(X_i) = \lambda \alpha e^{X_i} - 1$ if $X_i < 0$

4. EXPERIMENTAL SETUP AND RESULTS

This section discusses the environment used for conducting the experiments and the results obtained from the experiments. For this work, we implemented the algorithms and procedures as Python programs using the Librosa, Keras, and Tensorflow libraries. We used librosa to implement the MFCC extraction algorithm, while Keras and Tensorflow were used for the CNN implementation. Figure 6 shows spoken utterances of Hindi digits spoken by a single speaker. We can observe that each spoken word has its waveform structure, from which distinct features can be extracted. We built models with different activation functions, as previously mentioned. Table 3 shows the performance of the CNN model for different combinations of the datasets and activation functions. The present study considers the Hindi digit dataset spoken by native Hindi speakers, non-native speakers, and combined speakers. This is to understand the influence of the accent of a speaker on the recognition system. While understanding this influence, the optimal model is chosen by observing the performance of the CNN model with different activation functions.

The study considers activation functions Softplus, exponential linear unit (ELU), rectified linear unit (ReLU), scaled exponential linear unit (SELU), and Tanh in the input layers whereas Softmax and Softplus are in the output layer. For the Telugu speaker's spoken Hindi dataset, the model uses SELU in all the input layers and the Softplus function at the output layer gives the highest recognition rate of 88.44%. The model with SELU in all the input layers and Softmax at the output layer performs well for native Hindi speakers with a recognition rate of 95.15%. We observed that the Hindi-spoken digit model is optimal with the activation function SELU in the hidden layers and Softplus in the output layer. A hybrid model with RELU, ELU, RELU, and Softmax functions used at different layers gave the next highest performance on the combined dataset. The model has given the highest recognition rates of 88.44%, 95.15%, and 89.41% for non-native, native, and combined datasets respectively. The lowest recognition rate for the non-native Hindi dataset, native Hindi dataset, and the combined dataset is observed for the tanh function.

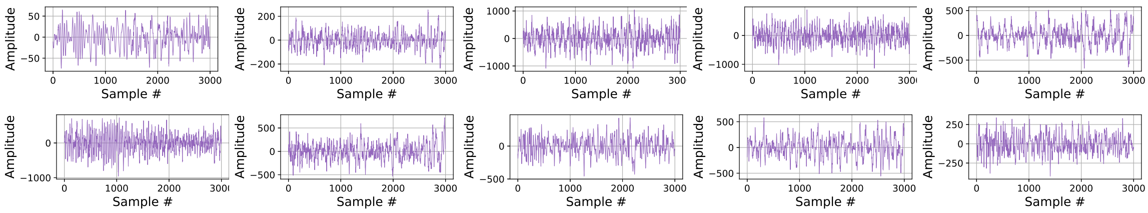

Figure 6. Hindi Spoken digits for a single speaker

Table 3. Performance of CNN for the training dataset with different activation functions

S. No	Activation function				Recognition rate (%)		
	Layer-1	Layer-2	Layer-3	Output	LBRCE	IITG	Combined
1	Softplus	Softplus	Softplus	Softmax	85.24	91.07	88.66
2	ELU	ELU	ELU	Softmax	87.37	92.09	88.42
3	SELU	SELU	SELU	Softmax	86.51	**95.15**	88.35
4	ELU	SELU	RELU	Softmax	87.61	94.64	87.46
5	RELU	RELU	RELU	Softmax	88.28	92.87	88.14
6	SELU	SELU	SELU	Softplus	**88.44**	93.36	**89.41**
7	Tanh	Tanh	Tanh	Softmax	71.98	86.73	75.71
8	RELU	ELU	RELU	Softmax	86.18	91.83	88.72
9	SELU	ELU	SELU	Softplus	84.09	94.38	87.87

5. CONCLUSIONS

In this paper, we investigated the CNN architecture to design the speech recognition model for the Hindi spoken data. The models used for recognizing the spoken data use MFCC as the major features to capture the essential patterns. Different activation functions were assessed to determine the most optimal activation function for the proposed model. The study also identified the appropriate activation functions for various layers. To test the model, the Hindi spoken digit dataset was used, collected from speakers belonging to different regions of India. The results showed that the proposed model efficiently recognized digits with a decent recognition rate for each digit.

REFERENCES

[1] B. M. L. Srivastava *et al.*, "Interspeech 2018 low resource automatic speech recognition challenge for Indian languages," in *6th Workshop on Spoken Language Technologies for Under-Resourced Languages (SLTU 2018)*, 2018, doi: 10.21437/SLTU.2018-3.

[2] P. Bhagath, S. Parisa, S. D. Reddy and F. Banu, "An Android based Mobile Spoken Dialog System for Telugu language to control Smart appliances," *2021 IEEE XXVIII International Conference on Electronics, Electrical Engineering and Computing (INTERCON)*, Lima, Peru, 2021, pp. 1-4, doi: 10.1109/INTERCON52678.2021.9532783.

[3] J. H. Tailor, R. Rakholia, J. R. Saini, and K. Kotecha, "Deep learning approach for spoken digit recognition in Gujarati language," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 4, 2022, doi: 10.14569/IJACSA.2022.0130450.

[4] N. Dalsaniya, S. H. Mankad, S. Garg, and D. Shrivastava, "Development of a novel database in Gujarati language for spoken digits classification," *Communications in Computer and Information Science*, vol. 1209, 2020, doi: 10.1007/978-981-15-4828-4_18.

[5] A. Dixit, A. Vidwans, and P. Sharma, "Improved MFCC and LPC algorithm for bundelkhandi isolated digit speech recognition," in *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, 2016, pp. 3755–3759, doi: 10.1109/ICEEOT.2016.7755413.

[6] P. Mohanty, J. P. Sahoo, and A. K. Nayak, "Voiced odia digit recognition using convolutional neural network," *Advances in Distributed Computing and Machine Learning*, vol. 302, pp. 161–173, 2022, doi: 10.1007/978-981-16-4807-6_16.

[7] R. Sharmin, S. K. Rahut, and M. R. Huq, "Bengali spoken digit classification: A deep learning approach using convolutional neural network," *Procedia Computer Science*, vol. 171, pp. 1381–1388, 2020, doi: 10.1016/j.procs.2020.04.148.

[8] G. Muhammad, Y. A. Alotaibi, and M. N. Huda, "Automatic speech recognition for Bangla digits," in *ICCIT 2009 - Proceedings of 2009 12th International Conference on Computer and Information Technology*, 2009, doi: 10.1109/ICCIT.2009.5407267.

[9] P. Bhagath, M. Pullagura, P. K. Das, V. K. Yandra, and S. S. Thetla, "Telugu ankelu: A telugu spoken digits corpora for mobile speech recognition," in *2022 12th International Conference on Pattern Recognition Systems (ICPRS)*, 2022, pp. 1–6, doi: 10.1109/ICPRS54038.2022.9854065.

[10] P. Bhagath, M. Pullagura, and P. K. Das, "Comparative analysis of spoken telugu digits using MFCC and LPCC via hidden markov models," in *2023 10th International Conference on Signal Processing and Integrated Networks (SPIN)*, Mar. 2023, pp. 615–620, doi: 10.1109/SPIN57001.2023.10116567.

[11] P. Bhagath, A. U. M. Rao, B. S. Ram, and M. A. K. Reddy, "Telugu spoken digits modeling using convolutional neural networks," in *2023 IEEE 13th International Conference on Pattern Recognition Systems (ICPRS)*, 2023, pp. 1–7, doi: 10.1109/ICPRS58416.2023.10178989.

[12] P. Bhagath and P. K. Das, "Quadrilaterals based phoneme segmentation technique for low resource spoken languages," in *TENCON 2022 - 2022 IEEE Region 10 Conference (TENCON)*, Nov. 2022, pp. 1–6, doi: 10.1109/TENCON55691.2022.9977455.
[13] A. Kumar and R. K. Aggarwal, "Discriminatively trained continuous Hindi speech recognition using integrated acoustic features and recurrent neural network language modeling," *Journal of Intelligent Systems*, vol. 30, no. 1, 2020, doi: 10.1515/jisys-2018-0417.
[14] J. Sangeetha, D. Rekha, M. Priyanka, and M. Dhivya, "An enhanced real-time automatic speech recognition system for Tamil language using wav2vec2 model," in *Advanced Applications in Osmotic Computing*, 2024, pp. 311–330, doi: 10.4018/979-8-3693-1694-8.ch016.
[15] B. Saxena and C. Wahi, "Hindi digits recognition system on speech data collected in different natural noise environments," in *International Conference on Computer Science, Engineering and Information Technology (CSITY 2015)*, 2015, pp. 23–30, doi: 10.5121/csit.2015.50303.
[16] I. Hussain and P. Roy, "A hybrid adaptive neuro-fuzzy approach for automatic spoken digit recognition," *International Journal of Speech Technology*, vol. 26, no. 4, pp. 825–832, 2023, doi: 10.1007/s10772-023-10057-6.
[17] A. S. M. B. Wazir and J. H. Chuah, "Spoken Arabic digits recognition using deep learning," in *2019 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS)*, 2019, pp. 339–344, doi: 10.1109/I2CACIS.2019.8825004.
[18] N. Zerari, S. Abdelhamid, H. Bouzgou, and C. Raymond, "Bi-directional recurrent end-to-end neural network classifier for spoken Arab digit recognition," in *2018 2nd International Conference on Natural Language and Speech Processing (ICNLSP)*, 2018, pp. 1–6, doi: 10.1109/ICNLSP.2018.8374374.
[19] R. Amari, A. Mars, and M. Zrigui, "Arabic speech recognition based on a CNN-BLSTM combination," in *2022 IEEE 9th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT)*, 2022, pp. 259–264, doi: 10.1109/SETIT54465.2022.9875681.
[20] X.-D. Wang, K. Owa, and M. Shozakai, "Mandarin digit recognition assisted by selective tone distinction," in *Interspeech 2010*, 2010, pp. 857–860, doi: 10.21437/Interspeech.2010-290.
[21] O. Abdel-Hamid, A. R. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 22, no. 10, pp. 1533–1545, 2014, doi: 10.1109/TASLP.2014.2339736.
[22] X. Li and Z. Zhou, "Speech command recognition with convolutional neural network," *Stanford Journal*, vol. 31, pp. 1–6, 2016.
[23] P. H. Pereira, W. Beccaro, and M. A. Ramirez, "Evaluating robustness to noise and compression of deep neural networks for keyword spotting," *IEEE Access*, vol. 11, pp. 53224–53236, 2023, doi: 10.1109/ACCESS.2023.3280477.
[24] Y. A. Wubet, D. Balram, and K. Y. Lian, "Intra-native accent shared features for improving neural network-based accent classification and accent similarity evaluation," *IEEE Access*, vol. 11, pp. 32176–32186, 2023, doi: 10.1109/ACCESS.2023.3259901.
[25] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "Activation functions in deep learning: A comprehensive survey and benchmark," *Neurocomputing*, vol. 503, pp. 92–108, 2022, doi: 10.1016/j.neucom.2022.06.111.

BIOGRAPHIES OF AUTHORS

Dr. Parabattina Bhagath is a Professor in the Department of Computer Science and Engineering at Lakireddy Bali Reddy College of Engineering in Mylavaram, India. He earned his M.Tech. and Ph.D. from the Indian Institute of Technology Guwahati, India in 2010 and 2020, respectively. He specializes in speech processing, particularly for resource-constrained languages and IoT devices. He has developed structural processing methods to understand speech signals using their structural properties. These procedures serve as alternative methods and procedures to address low-resource data, in addition to conventional machine-learning techniques. His research has been published in the proceedings of various esteemed conferences. He can be contacted at email: bhagath.2014@alumni.iitg.ac.in.

Malempati Shanmukha completed B.Tech. specializing in the Department of Computer Science and Engineering at Lakireddy Bali Reddy College of Engineering, located in Mylavaram, India. His academic pursuits are currently focused on the field of speech processing, a branch of computational linguistics and artificial intelligence that deals with the interaction between computers and human languages through spoken words. In his research, he has developed language models for various spoken units in multiple languages, including Telugu, English, and Hindi. His work involves analyzing the structure and patterns of spoken language and creating algorithms that can process and interpret speech effectively. He can be contacted at email: malempatishanmukha@gmail.com.

Prof. Pradip K. Das is a Professor in the Department of Computer Science and Engineering at the Indian Institute of Technology Guwahati, India. He is the Head of the Centre for Drone Technology and working in the areas of unmanned aerial vehicles, speech processing, IoT, and bio-medical signal processing. He completed his Ph.D. from the University of Delhi in 1999. He has been associated with IIT Guwahati from the year 1998 holding different positions such as Assistant Professor and Associate Professor. He has been working on various funding projects of different government agencies including ONGC, MHRD, DIT, and CDAC. He can be contacted at email: pkdas@iitg.ac.in.