2022 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON) | 978-1-6654-6374-4/22/$31.00 ©2022 IEEE | DOI: 10.1109/CENTCON56610.2022.10051465

A Comparative Study of Feature Modelling Methods for Telugu Language Identification

Mandava Jaswanth
Department of Computer Science and Engineering
Amrita School of Computing
Bengaluru, India
jmandava027@gmail.com

Namburi K L Narayana
Department of Computer Science and Engineering
Amrita School of Computing
Bengaluru, India
narayananamburi111@gmail.com

Sreedharreddy Rahul
[1]Department of Computer Science and Engineering
Amrita School of Computing
Bengaluru, India
rahulreddy1109@gmail.com

Susmitha Vekkot
Department of Electronics and Communication
Amrita School of Engineering
Bengaluru, India
v_susmitha@blr.amrita.edu

Sreedharreddy Rahul
Department of Computer Science and Engineering
Amrita School of Computing
Bengaluru, India
g_deepa@blr.amrita.edu

Abstract—**Telugu is one of the most widely spoken language, with 82.7 million native speakers. Telugu is used predominantly in the Indian states of Telangana and Andhra Pradesh. This paper explains the design and implementation of a model that identifies whether the language spoken by the user is Telugu or from a multilingual dataset containing audio clips from Telugu, Malayalam, Gujarati, English, and Marathi of native Indian speakers. Five features, MFCC, Chroma CQT, Chroma STFT, Chroma CENS, Tonnetz-space, and combinations of these, are used for training the models. Five powerful deep learning architectures CNN, LSTM, CNN-LSTM, BiLSTM, and GRU are used in training over the features and different feature combinations. The performance of the models has been compared with state-of-the-art models for language identification. The proposed models yielded maximum accuracy of 94% for CNN with Chroma CQT+MFCC feature combination.**

Keywords— Long Short-Term Memory, Telugu, Language Identification, Gated Recurrent Units, Deep Neural Networks, Speech Processing, Convolutional Neural Networks.

I. Introduction

The identification of different languages remains to be one of the challenging problems in the area of speech-language identification. The problem is in determining the identity of a language by an anonymous source of the voice. It becomes more complicated when one needs to work by using short, informal, and almost the same language voice clips. These types of cases are complex and primarily found in the content of social media. In today's tech world, language identification systems can help us communicate, such as speech and document translations from one language to another. The usage of language recognition technology and personal assistants has substantially changed technology. They have moved from mobile phones to homes, and their use and government sectors also have increased [1]. Voice assistants like Google assistant, Siri, and Alexa use language identification. These voice assistants respond to the language that the user speaks. They classify language by listening to what people speak and responding to us. In real-life scenarios, many applications, such as call center customer routings and multi-language identification systems, are made possible through language recognition. Model accuracies can be improved using features and deep learning techniques, but the research on Telugu language identification is limited. Telugu is a language that enjoys the position of being the most spoken classical and Dravidian language in India. It is the official language of more than one Indian State. Telugu has importance as an old language of philosophy and literature. Due to these reasons, Telugu is rich in resources, and the availability of a large amount of speech data for modeling is a prominent reason for adopting Telugu to deep learning applications. Language identification systems can be expanded to applications involving all Dravidian languages and can be used in other scenarios wherein resources may be scarce or limited. Therefore, research detailing feature modeling for Telugu data will be impressive. The main contributions of work in this paper are:

- Feature modeling with audio clips using individual and mixed features viz Mel-frequency cepstral coefficients (MFCC), Constant-Q chromagram (CQT), Chroma short-term Fourier transformation (STFT), Chroma Energy Normalized (CENS), Tonnetz-space.
- A combination of features: CENS+MFCC, CENS+Tonnetz, CQT+MFCC, CQT+CENS, CQT+Tonnetz, STFT+MFCC, STFT+CENS, STFT+Tonnetz, Tonnetz+MFCC, STFT+CQT are also used for modelling
- Performance comparison of Telugu language identification models built using the deep neural networks viz Convolutional Neural Network (CNN), Long Short-Term Memory Networks (LSTM), Bidirectional recurrent neural networks with Attention Layer (BiLSTM), Gated Recurrent Unit (GRU), Convolutional Long Short-Term Memory Networks (CNN-LSTM).

II. Related Work

Researchers in [1] developed a model based on recurrent neural networks, which is bi-directional and performs better in monolingual and multilingual language detection functions on diverse datasets comprising 131 languages. The authors used multiple models like Hidden Markov Model, Random Forest, and Naïve Bayes classifiers to identify multilingual data [2]. A hierarchical model for learning

978-1-6654-6374-4/22/$31.00 ©2022 IEEE

character and contextualized [3] word-level representations for language recognition outperforms solid baselines and reveals code-switching [4]. The method developed by the authors [5] can be used in various noisy situations and can be easily expanded by including undiscovered languages when retaining classification accuracy [6]. The authors look at combining a naive Bayes model with a lexicon-based classification for reliably predicting the family to which the small text has belonged, as well as the classifier based on lexicon to identify particularly the languages [7] of South Africa. The method yields different variations in the detection of language error reduction [8,9]. Authors in [10] used deep convolutional neural networks as discriminative models to examine the challenge of spoken language recognition for closely related languages in a cross-domain setting[11]. The authors offer a deep neural network-based language identification technique that achieves near-perfect accuracy when categorizing different languages and about ninety percent when classifying similar languages [12]. Researchers have used various deep learning architectures [13] for image classification to detect languages from audio and video-produced pictures. Using small files with little pre-processing resulted in powerful performance [14]. The transformation scales must be developed to represent the contribution of each emotion and are independent of linguistic features to create a multilingual model [15]. The feature extraction and classification modules comprise the bulk of the speech emotion recognition (SER) method. Pioneers in the field of speech processing have recently developed a variety of speech characteristics, including hybrid features, vocal tract parameters, prosodic features, and excitation source features, for this purpose [16]. The authors in [17] used several languages and assessment methodologies to achieve automated language recognition in texts and voices. Neural network-based language recognition and user identification solution are presented in [18]. The technique is based on the sequential minimal optimization learning method for SVM that improves performance over traditional approaches [18]. The authors demonstrated a method that takes in NPF and MFCC features as the speech signal characteristics, and artificial neural networks are used as the feature mapping technique [19].

The survey acknowledges that research on multilingual language detection problems contains three significant challenges: model selection, feature identification, and optimization of training. In state-of-the-art language identification systems, only MFCC and spectrogram are used as features. The performance of the above models can be improved by combining spectra and Mel cepstra features. The work performed in this paper carries forward this investigation for the Telugu language using state-of-the-art systems in deep learning.

III. Methodology

The primary approach for identifying the Telugu language is multiclass classification using different deep-learning techniques involving CNN, LSTM, bi-LSTM, CNN-LSTM, and GRU. Fig. 1 depicts the process flow diagram of the modeling strategy used in work. The feature descriptions and the details on the main modules of the model are discussed in the following subsections.

A) Dataset

The dataset used for language identification, as given in Table I, contained short audio clips of males and females of five languages – Telugu, Malayalam, Gujarati, English, and Marathi. The dataset size is around 7.0 GB, in which there are 4000 Telugu language wav files and 4000 wav files from the remaining languages, which are Malayalam, Gujarati, English, and Marathi.

TABLE I. DESCRIPTION OF LANGUAGE IDENTIFICATION DATASET

Attributes	**Male**	**Female**	**Total**
Avg. Duration of each sample	7 secs	7 secs	7 secs
Number of Samples	4323	3677	8000
Number of Telugu Language Samples	2126	1874	4000
Number of Other Language Samples	2197	1803	4000

B) Feature Extraction

Five spectral features: MFCC, Chroma STFT, Chroma CENS, Tonnetz-space, and Chroma CQT are extracted and used for model training. This is to provide identifiable information and representations that aid in a more accurate classification of audio signals. Different features are extracted using the librosa python library. In addition to individual features, various feature combinations are also used for modeling, as detailed in section IV.

1) MFCC

MFCC acts as a typical feature while working with audio data. These coefficients represent an audio signal and can be dynamically used to recognize speech, as their significance is high in language recognition. The process of extracting MFCC features includes computation of the discrete Fourier transforms of the windowed speech signal, calculating the log magnitude and frequencies warped on the Mel scale, and finally, using the inverse discrete cosine transforms.

2) Chroma STFT

Audio chroma values represent the strength of 12 pitch classes used in music research. These chroma values can differentiate pitch profiles between audio signals [20].

3) Chroma CENS

This technique calculates all the statistical data, like tempo, articulation, arpeggios, etc., using large windows. Chroma attributes exist based on the 12 pitch characteristics from western musical notation. Each chroma variable shows the distribution of audio intensity across the 12 chroma bands [20].

4) Chroma CQT

Another chromagram technique is the chroma constant Q (CQT) transformation. In this technique, STFT values will be used, and different techniques will be applied to achieve these values; the techniques involve increasing the differentiation from the frequency bins. However, mapping the frequencies to a logarithmic scale presents problems since the bin size is fixed for all frequencies. To address this issue, the main goal of CQT is to decrease the buffer size for significant frequencies and increase the buffer size [21] when the frequencies are low; thereby reducing the computation load.

5) Tonnetz-space

This model translates 12-bin chroma vectors to the internal spaces of a 6-D polytope; pitch classes are mapped onto the polytope's vertices. Close harmonic relationships, such as fifths and thirds, appear as short Euclidian distances. To construct a harmonic change measure for frame n, the Euclidian distance is calculated between signals from analysis frames n +1 and n -1. A peak in the detection function indicates a transition from one harmonically stable zone to another [22].

For our categorization method, spectral features were used. Which are as follows: MFCC, Chroma STFT, Chroma CQT, Chroma CENS, and Tonnetz-space[23]. Due to the specific information and representations they offer for describing the linguistic characteristics; audio signals can be classified more precisely. These features are extracted after exploratory data analysis and preprocessing, as shown in Fig-1.

C) Supervised Learning Models:

The extracted features and their corresponding labels are stored and divided into train and test features, as shown in Fig-The train features and labels are used to train the supervised learning models.

1. Long Short-Term Memory:

Long Short-Term Memory (LSTM) has recurrent connections, different from feedforward CNN. LSTM handles both single data points as well as actual data. First, at a fundamental level, the output of an LSTM at a certain moment depends on three factors:

- The current long-term memory of the network, also known as the cell state.
- The prior output, also known as the previous hidden state
- The data entered at the current time step.

The two LSTM layers are piled up in our model. Since the recurrent sequence has been put to "True," and there are 128 hidden units in the first LSTM layer and only 64 in the second, both LSTM layers need to output a 3D array that will be used as input to the dense layer. The input size for our model's first layer is (a, b), where ‘a’ stands for time steps that inform the LSTM layer how frequently it should occur after applying it to the input. Overfitting is decreased by using a dropout of 0.3.

2. Gated Recurrent Units(GRU):

Gated recurrent units have forget gate, but it does not have an output gate; thus, it has fewer parameters than LSTM. On tasks involving music, speech signal modeling, and NLP, it was found that the performance of GRU could occasionally match that of LSTM. GRU performs better on smaller and less frequent datasets [24].

Two GRU layers are piled up in our model. Since the recurrent sequence has been put "True" and there are 128 hidden units in the first GRU layer and only 64 in the second, both GRU layers must produce a 3D array that will be used as input to the denser layer. The input size for the first layer of our model is (a, b), where ‘a’ denotes time steps, which tell a GRU layer how many times it should occur after applying it to the input. A dropout layer with a size of 0.3 was used to reduce the overfitting of the model.

3. Convolutional Neural Networks(CNN):

CNN uses the inputs as images and can assign significance in the form of learnable weights and biases to many different features in the image. CNN requires lesser pre-processing compared to other classification algorithms. While simple techniques need hand-engineering of filters, with enough training, CNN can learn these filters/characteristics.

The first Convolution 2D layer [25], which receives the input shape, includes 64 filters with a kernel size of 5 and stride of 1. A (5x5) filter matrix is the outcome of the parameter's specification of the kernel window's size. The filter goes one unit simultaneously as it converges around the input volume in the first layer, where the stride is set to 1. The height and weight of the output from this convolutional layer with the same padding operation match those of the input. ReLU is used as the activation function for this layer, due to the advantages of efficient gradient propagation and faster calculation, compared to sigmoid units [19]. The initial Conv2D layer is followed by the MaxPooling2D layer, which reduces the input shape dimension. In the second Conv2D layer, ReLU acts as the activation function. This layer has 128 filters, a kernel of size 5, a stride of 1, and the same padding operation. A 30% dropout is used in the second Conv2D layer to reduce overfitting.

4. CNN-LSTM Network:

CNN and LSTM layers are used for feature extraction and sequence prediction in the CNN-LSTM architecture [26]. This architecture is used to generate textual image descriptions.

To extract various features, LSTM and CNN models are piled up; two convolution 2D layers, one convolution layer, one max pooling layer, a batch normalization layer, flatten layer, a dense layer, and a drop-out layer are used. This output is fed to the LSTM model indicated above. Finally, a softmax activation function was used to forecast the outcome.

5. Bi-directional LSTM:

Bi-directional long short-term memory (Bi-LSTM) is a kind of recurrent neural network primarily used to process natural language. Unlike traditional LSTM, it can use data from both sides and has a two-way input flow. Furthermore, it is an effective tool for modeling the sequential dependencies between words and phrases in both the forward and backward directions of the sequence.

IV. IMPLEMENTATION

Initially, data collection is performed from google open speech and language resources, with properties as discussed in Table I. Exploratory data analysis is conducted for the dataset, followed by data pre-processing, feature extraction train:test split, training, and k-fold cross-validation, the details as discussed in the following subsections.

A. Exploratory Data Analysis

In speech or audio data, there can be two different dependencies frequency dependent and temporal-based. A spectrogram depicts how loud or strong a signal is overtime at various frequencies present in a particular waveform. In addition to evaluating if, for example, there is energy at 2 Hz compared to 10 Hz, one may also see how energy levels fluctuate over time. To analyze this audio data, a spectrogram is used. Data analysis involves measuring the spectral centroid, roll-off, spectral bandwidth, and zero-crossing rate to analyze the signal's shape and understand the frequency upon which the energy spectrum is centered. In language recognition, the input can be either spectrogram or more complex features like MFCC, Chroma-CQT, etc. An exploratory data analysis is performed to identify the features to train, for data tuning and enlightening elements in process of feature selection [27]. Visualization of the spectrogram helps in identifying noise unusual silences. Silence removal is performed where there is no activity for more than 5 seconds after visualization of the audio spectrogram. These steps were carried out to eliminate unwanted features, minimizing the number of features [28].

B. *Data pre-processing and Feature extraction*

Initially, to make the audio duration the same, 2-3 sec silences are added to the beginning of the audio files, thereby making it easy for windowing. Then, the temporal segments are mapped to their language labels (Telugu or other). Feature extraction is performed using Python libraries. Data is then split into 80:20 train-test ratio.

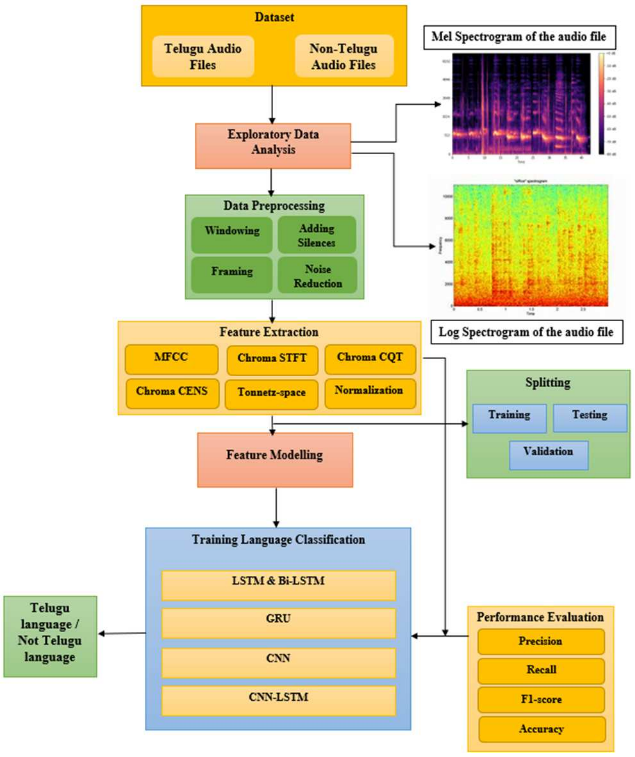

Fig 1: Process Flow Diagram

C. *Training*

The preprocessed data is split into training, testing, and validation sets and used to train and validate our model to avoid overfitting, and five-fold cross-validation is used here. Table II gives the performance of each model using individual feature training.

V. Results and Discussion

In this experiment, all five models are evaluated based on different features for different model combinations and the performance is analyzed. As mentioned in Table II, the maximum accuracy achieved through CNN modeling is 89.2. using the MFCC feature. Training with LSTM, the maximum accuracy achieved is 92.89, precision is 90%, F1-score is 0.91, and recall is 0.91 using the same features. This accuracy is the highest in all the models and features; language models aim to anticipate the subsequent word based on the words that came before it. Given that BiLSTM also anticipates future words, it is inappropriate. Poor accuracy will result from using BiLSTM in this application. Furthermore, BiLSTM takes longer than LSTM since the forward pass results must be accessible before the backward pass can begin. Therefore, gradients will have a long chain of dependencies. Further, it is observed that all the other models also give good accuracies with the MFCC feature alone, so from individual features analysis, it can be concluded that MFCC is the best feature to interpret language. Table II shows the individual five-fold cross-validation values for each model.

TABLE II. PERFORMANCE OF ALL DEEP LEARNING ARCHITECTURES ON VARIOUS FEATURES

Models	Evaluation Metrics	MFCC	Chroma STFT	Chroma CQT	Chroma CENS	Tonnetz-space
CNN	Accuracy	89.2	87.02	87.78	84.02	85.51
	Precision	0.88	0.9	0.86	0.83	0.85
	F1-score	0.87	0.88	0.86	0.823	0.86
	Recall	0.88	0.86	0.87	0.85	0.85
LSTM	**Accuracy**	**92.89**	80.95	80.85	77.38	68.46
	Precision	**0.9**	0.81	0.81	0.75	0.7
	F1-score	**0.91**	0.81	0.79	0.76	0.69
	Recall	**0.91**	0.82	0.79	0.75	0.69
GRU	Accuracy	90.18	78.53	79.23	75.55	66.97
	Precision	0.89	0.77	0.78	0.73	0.63
	F1-score	0.9	0.77	0.76	0.72	0.64
	Recall	0.9	0.78	0.78	0.71	0.66
CNN-LSTM	Accuracy	89.44	80.17	81	75.17	71.9
	Precision	0.88	0.81	0.83	0.74	0.72
	F1-score	0.86	0.81	0.82	0.73	0.7
	Recall	0.87	0.81	0.81	0.73	0.72
Bi-LSTM	Accuracy	89.49	81.63	82.12	78.02	69.72
	Precision	0.9	0.81	0.82	0.76	0.69
	F1-score	0.89	0.81	0.81	0.77	0.68
	Recall	0.9	0.81	0.82	0.77	0.68

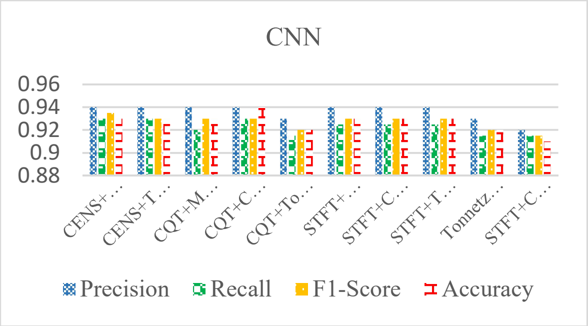

Fig-2: Combined feature performance of CNN

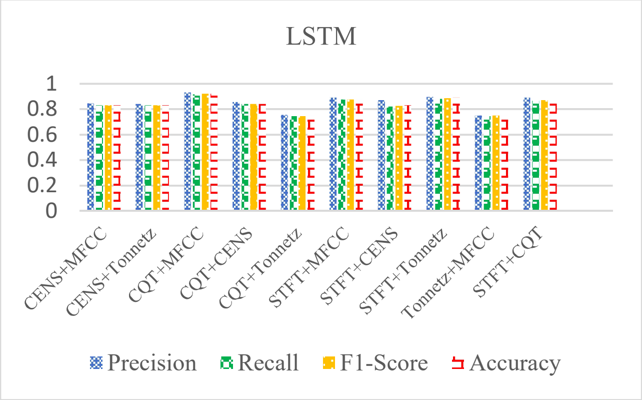

Fig-3: Combined feature performance of LSTM

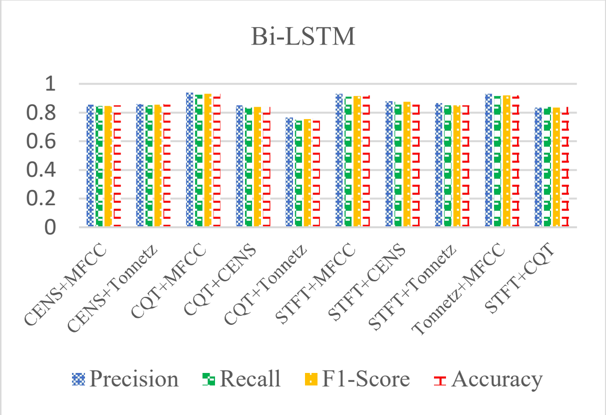

Fig-4: Combined feature performance of Bi-LSTM

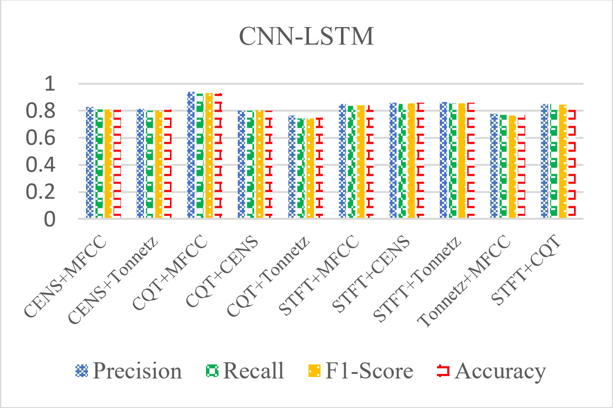

Fig-5: Combined feature performance of CNN-LSTM

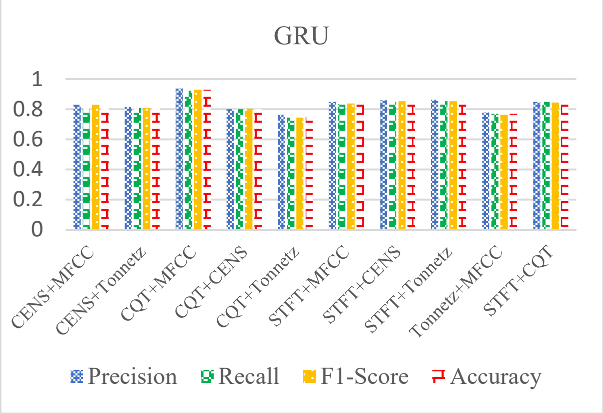

Fig-6: Combined feature performance of GRU

Another aspect that we have considered in this work is model performance analysis when the features are combined in various orders. For example, if two features are taken into consideration, viz. MFCC and Chroma STFT, a combination of both features is performed and appended into a single feature vector which in turn is used to train all the models. The results are displayed in Fig-2 - Fig-6.

From the figures, it is inferred that when the features are combined, the results are more consistent, and each model improved its accuracy, precision, recall, and F1 score. Fig-2 shows that the CNN model performed much better with the feature combination Chroma CQT + MFCC. The same feature combination performed well with all models and achieved good accuracy compared to all other features discussed in this paper. Therefore, training using combined spectral and cepstral features increased the efficacy of Telugu language identification.

By observing the results of the performance indicators, three combinations with MFCC viz. Chroma CQT+MFCC, STFT+MFCC, and CENS+MFCC performed well for the five models. As we observed, MFCC performed well as a single feature itself, and further, when combined with other Chroma features, it gave better results. It is also interesting to note that features like Chroma CENS, Chroma STFT, and Tonnetz, which gave less performance indicator values (accuracy – 85%) when taken individually, gave an accuracy of around 93% when combined. CENS+MFCC, CENS+Tonnetz, CQT+MFCC, CQT+CENS, CQT+Tonnetz, STFT+MFCC, STFT+CENS achieved an accuracy of 94%, precision of 93%, with recall and F1-scores of 92% and 92% respectively. Tonnetz+MFCC and STFT+CQT achieved an accuracy of 93%, 94% precision, 93% recall, and F1-score with CNN.

In Table III, the comparison using various models is depicted with state-of-the-art but with different features, languages, and models. However, there is a chance to improve the model and accuracy by using various other features. Also, most of the research on language recognition is done in English. For Telugu language identification, our proposed features with LSTM can achieve an accuracy of 93.89%. In contrast, the same for the language identification model in English is 98.70 using temporal features with 2D ConvNet with Attention and GRU.

VI. Conclusion And Future Scope

The research in this paper focused on developing a language identification model for Telugu using a multilingual dataset, it is observed that LSTM worked best with the MFCC features, giving an accuracy of 92.89%. CNN performed best with a combination of Chroma CQT and MFCC features, yielding an accuracy of 94%. The results are promising and demonstrate the effectiveness of CNN and LSTM on combined spectral & cepstral feature

TABLE III. COMPARATIVE ANALYSIS OF THE PROPOSED SYSTEM WITH STATE-OF-THE-ART

Year	Model(s) used	Language	Features Involved	Language Identification	Accuracy	Dataset
2019	VGG16 and Bernoulli naïve bayes [13]	Mixed	Spectral features	Yes	93%	LRE and Google 5M
2019	2D ConvNet with Attention and GRU [24]	Mixed	Spectral features using CNN and RNN	Yes	93.70%	Voxforge
2020	SLR [16]	Mixed	X-vector embeddings	Yes	-	VoxLingua107
2021	CRNN with Attention [23]	Mixed	Temporal Features	Yes	98.70%	LDC SLR dataset
2022	TDNN [14]	Mixed	MFCC	Yes	66.33%	Media Speech
Proposed Model	**Custom with CNN**	**Telugu**	**MFCC, CQT, STFT, CENS, Tonnetz-space, and combinations of these features.**	**Yes**	**94%**	**Open SLR**

There is space for enhancement by balancing, enhancing the data with pitch shifting, cropping, rotating, and flipping, as well as by adding random noise and modifying audio speed. These data pre-processing methods make neural networks more resistant to alterations that could take place in actual-world situations. It is possible to monitor or analyze different feature extraction techniques, such as zero crossing rate, and their impact on language recognition. Deep neural networks have performed better because of these settings. Voice recognition technology may be incorporated into business organizations' digital plans, marketing strategies, and budgets.

REFERENCES

[1] K. Tom, and O. Bojar, "Lanidenn: Multilingual Language Identification on Character Window," In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics., vol.1, pp.927–936, Apr 2017.

[2] S. Gundapu, and R. Mamidi, "Word Level Language Identification in English Telugu Code Mixed Data," In Proceedings of the 32nd Pacific Asia Conference on Language, Information, and Computation., 2018.

[3] A. Jaech, G. Mulcaire, S. Hathi, M. Ostendorf, and Noah A. Smith, "Hierarchical character-word models for language identification," In Processings of the 4th International Workshop on Natural Language Processing for social media., pp.84-93, Nov 2016.

[4] S. Vekkot, D. Gupta, M. Zakariah, and Y. A. Alotaibi, "Emotional voice conversion using a hybrid framework with speaker-adaptive DNN and particle-swarm-optimized neural network," IEEE Access, 8, 74627-74647, 2020.

[5] C. Bartz, T. Herold, H. Yang, and C. Meinel, "Language Identification Using Deep Convolutional Recurrent Neural Networks," In International conference on neural information processing., pp. 880-889, Nov 2017.

[6] S. Vekkot & D. Gupta, "Fusion of spectral and prosody modelling for multilingual speech emotion conversion," Knowledge-Based Systems, 242, pp. 108360, 2022

[7] Duvenhage, Bernardt, M. Ntini, and P. Ramonyai, "Improved text language identification for the South African languages," In 2017 Pattern Recognition Association of South Africa and Robotics and Mechatronics (PRASA-RobMech)., pp. 214-218, Nov 2017.

[8] B. Duvenhage, "Short text language identification for under resourced languages," arXiv preprint arXiv:1911.07555, 2019.

[9] Marwa A. Nasr, M. Abd-Elnaby, Adel S. El-Fishawy, S. El-Rabaie, and Fathi E. Abd El-Samie, "Speaker identification based on normalized pitch frequency and Mel Frequency Cepstral Coefficients," International Journal of Speech Technology., vol.4, pp.941-951, Dec 2018.

[10] B. Abdullah, T. Avgustinova, B. Möbius, and D. Klakow, "Cross-Domain adaptation of Spoken Language Identification for Related Languages: The Curious Case of Slavic Languages," International Speech Communication Association (ISCA)., Sep 2020.

[11] P. Mathur, A. Misra, and E. Budur," Language Identification from Text Documents," arXiv:1701.03682v1, Jan 2017.

[12] S. Vekkot, and D. Gupta, "Speaker-independent expressive voice synthesis using learning-based hybrid network model," International Journal of Speech Technology., vol.3, pp.597-613.

[13] G. Singh, S. Sharma, V. Kumar, M. Kaur, M. Baz, and M. Masud," Spoken Language Identification Using Deep Learning," Hindawi Computational Intelligence and Neuroscience., Sep 20121.

[14] T. Jauhiainen, M. Lui, M. Zampieri, T. Baldwin, and K. Linden, "Automatic Language Identification in Texts: A Survey," Journal of Artificial Intelligence Research, 2018.

[15] S. Lalitha, S. Tripathi, and D. Gupta. "Enhanced speech emotion detection using deep neural networks," International Journal of Speech Technology., pp.497-510, 2019.

[16] S. Vekkot, and D. Gupta. "Prosodic transformation in vocal emotion conversion for multi-lingual scenarios: A pilot study." International Journal of Speech Technology., pp.533-549, 2019.

[17] S. M. Kamruzzaman, A. N. M. Rezaul Karim, Md. Saiful Islam and Md. Emdadul Haque, "Speaker Identification using MFCC-Domain Support Vector Machine," International Journal of Electrical and Power Engineering., vol.1, pp.274-278, 2007.

[18] J. Valk, and T. Alumäe, "VoxLingua107: A Dataset for Spoken Language Recognition," 2021 IEEE Spoken Language Technology Workshop (SLT)., pp.652-658, 2021.

[19] B. Kepecs, and H. Beigi, "Automatic Spoken Language Identification using a Time-Delay Neural Network," arXiv preprint arXiv:2205.09564., May 2022.

[20] A. K. Shah, M. Kattel, A. Nepal, and D. Shrestha, "Chroma feature extraction." Conference: chroma feature extraction using Fourier transform., vol.20, 2019.

[21] K. O'Hanlon and M. B. Sandler, "Comparing CQT and Reassignment Based Chroma Features for Template-based Automatic Chord Recognition," ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)., pp. 860-864, 2019.

[22] C. Harte, M. Sandler, and M. Gasser, "Detecting Harmonic Change in Musical Audio," In Proceedings of the 1st ACM Workshop on Audio and Music Computing Multimedia., pp. 21-26, Oct 2006.

[23] A. Ali, N. Dehak, P. Cardinal, S. Khurana, S. Harsha Yella, J. Glass, P. Bell, and S. Renals, "Automatic Dialect Detection in Arabic Broadcast Speech," Proc. Interspeech 2016., pp.2934-2938, 2015.

[24] Sarthak, S. Shukla, and G. Mittal. "Spoken Language Identification using ConvNets," European Conference on Ambient Intelligence., pp.252-265, 2019.

[25] S. Vekkot, D. Gupta, M. Zakariah, and Y. A. Alotaibi, "Hybrid Framework for Speaker-Independent Emotion Conversion Using i-Vector PLDA and Neural Network," in IEEE Access, vol. 7, pp. 81883-81902, 2019.

[26] A. Mandal, S. Pal, I. Dutta, M. Bhattacharya, and S. Naskar, "Is attention always needed? A Case Study on Language Identification from Speech," arXiv preprint., Oct 2021.

[27] M. Senbagavalli and T. Arasu, "Opinion Mining for Cardiovascular Disease using Decision Tree based Feature Selection," Asian Journal of Research in Social Sciences and Humanities., vol.6, Issue.8, pp no. 891-897, 2016.

[28] MS Valli, T. Arasu, "An Efficient Feature Selection Technique of Unsupervised Learning Approach for Analyzing Web Opinions," Journal of Scientific & Industrial Research, NISCAIR-CSIR., vol. 75, pp. 221-224, Apr 2016.