Journal of Information Systems Engineering and Management
2025, 10(5s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Comparative Analysis of CNN Architectures for English and Gujarati Speech Recognition Using MFCC Features

Jasmine J. Karagthala[1], Dr. Vrushank Shah[2]
[1]PhD Scholar Department of E& C, [2]Head of Department of E& C,
[1]Indus University, Ahmedabad, India [2]Indus University, Ahmedabad, India
Email id: [1]jdaftary@gmail.com, [2]ec.hod@indusuni.ac.in
[1]ORCHID: [1]0009-0006-4801-8885, [2]0000-0002-5619-7100

ARTICLE INFO

Received: 07 Oct 2024

Revised: 14 Dec 2024

Accepted: 26 Dec 2024

ABSTRACT

The paper investigates the efficiency of Convolutional Neural Network (CNN) architectures for speech recognition, focusing on the English and Gujarati languages. The study explores the impact of different CNN layer depths, utilizing 2, 3, and 4-layer configurations. Mel-Frequency Cepstral Coefficients (MFCC) is employed for feature extraction before feeding the data into the CNN models. The activation functions Rectified Linear Unit (ReLU) and hyperbolic tangent (tanh) are examined across all architectures. The research uses the Speech Commands dataset for English and a Gujarati digits dataset for analysis. After preprocessing and MFCC feature extraction, CNNs with varying depths and ReLU activation are employed. Training encompasses both languages, exploring parameters for balanced performance and efficiency, emphasizing tailored solutions for diverse linguistic contexts. The results reveal that the ReLU consistently yields superior performance on both the English and Gujarati datasets. In addition, the study found that increasing the depth of the CNN layers does not necessarily lead to improved recognition accuracy. The findings underscore the importance of selecting appropriate activation functions, highlight the nuanced relationship between CNN depth and recognition performance, and contribute to the understanding of CNN architecture optimization for speech recognition tasks in diverse linguistic contexts. The insights gained can inform the design of more effective speech recognition systems for globally recognized languages, such as English, and vernacular languages like Gujarati.

Keywords: Mel-Frequency Cepstral Coefficients (MFCC), Convolutional Neural Network (CNN) architecture, Rectified Linear Unit (ReLU), Speech Recognition (SR).

INTRODUCTION

Speech Recognition systems transform spoken language into text, enabling communication between humans and computers[1]. This conversion is achieved through algorithms that analyze acoustic signals, identify phonetic patterns, and match them with pre-existing linguistic models. Machine learning techniques, especially the ones based on deep neural networks, are essential to allow the system to learn the way humans speak and interpret, converting speech into writing [2][3]. There are two main types: speaker-dependent, which requires data from specific users; and speaker-independent, which works for anyone. Speech recognition is also classified based on the level it operates at, such as identifying individual sounds (phonemes), whole words, or complete sentences gujarati[4]. How accurate the recognition of speech is, depends on the type of speech, the size of the vocabulary, and the context of the use of the latter. One must also mention the significance of accents[5] and typical variations of pronunciation across people [6]. The fact that two different individuals may pronounce the same word in a manner sufficiently distinct to make it indiscernible complicates the use of phonological levels. MFCCs and spectrograms are used to turn speech into data[7][8]. Hidden Markov models and neural networks are then used to interpret the data and recognize patterns [9]. Hidden Markov models (HMMs) face challenges in capturing long-range dependencies in speech due to their inherent Markovian assumptions, which may limit their ability to model complex temporal relationships accurately[10]. While powerful, neural networks require large amounts of annotated data for training and may suffer from overfitting or generalization issues, particularly in the presence of noisy or variable input data. Additionally, both approaches may struggle with effectively handling variations in speaking styles, accents, and background noise.

Speech recognition, as discussed, has wide applications in various technical fields. It is used in automatic dictation systems that type medical reports, and legal documents and voice-controlled interfaces used in automotive navigation. Other practical applications of speech recognition include real-time language translation services. The applications have not only improved productivity but also sparked innovation. For instance, healthcare services have been transformed, and legal services have become more efficient. Various studies have been conducted in speech recognition to enhance their level of accuracy, robustness, and usability. This involves developing better feature extraction methods, such as MFCCs and spectrograms, and refining classification algorithms, such as hidden Markov models and neural networks [11]. Understanding the details of human speech and creating algorithms to manage differences is an ongoing challenge in this field[12]. Speech recognition is a complex interdisciplinary field that combines linguistics, signal processing, machine learning, and artificial intelligence Ongoing technological advancements and research are steadily enhancing the accuracy and usability of speech recognition, making it an essential part of how we interact with computers [13].

This paper recognizes the importance of both English and Gujarati language[14]. English, as a global language, facilitates international communication and research, while Gujarati, a regional language, is vital for preserving culture and communication among its many speakers. English is often used as a common language in various fields, and Gujarati, one of India's official languages, is spoken by over 50 million people [15]. The English language have its widespread use and the availability of standardized datasets like the Speech Command dataset from Kaggle, providing a solid basis for comparison[16] [17]. Similarly, Gujarati highlights the significance of regional languages in promoting inclusivity and accessibility, using datasets such as Gujarati spoken digits to test speech recognition in diverse linguistic settings[18] .This bilingual approach not only enhances the understanding of speech recognition across languages but also emphasizes the need for tailored solutions to address regional linguistic diversity[19] [20].

To extract features from audio data, mel-frequency cepstral coefficients (MFCC) were employed a proven method in speech recognition renowned for its efficacy in capturing phonetic nuances. These features were then fed into three distinct convolutional neural network (CNN) architectures: 2-layer, 3-layer, and 4-layer CNNs, each leveraging diverse activation functions to evaluate their impact on the model performance. This methodology enables an in-depth exploration of how variations in neural network depth and activation functions influence speech recognition accuracy across linguistically and culturally diverse languages. Key findings revealed that the rectified linear unit (ReLU) consistently outperformed the hyperbolic tangent (tanh) activation function across both language datasets. Surprisingly, deeper CNN architectures did not uniformly translate into improved performance, suggesting nuanced dependencies between the network complexity and recognition accuracy. Furthermore, it was observed that the English language dataset exhibited a richer set of results compared to its Gujarati counterpart. These insights shed light on the optimal model configurations for speech recognition tasks and underscore the importance of considering linguistic diversity in model development.

RESEARCH CONTRIBUTION

This research contributes to the advancement of speech recognition technology by conducting a comparative analysis of CNN architectures for English and Gujarati speech recognition using MFCC features. It solves the problem of building effective voice recognition systems by looking at both languages. In order to better understand how different CNN architectures work in diverse language settings, the research compares and contrasts their performance. The study also offers useful insights into the cross-lingual applicability of MFCC characteristics by using them, which are a standard in voice recognition. The research provides empirical information to help choose the best convolutional neural network (CNN) architectures for comparable voice recognition tasks in different language contexts by extensively evaluating their performance using measures such as accuracy and F1-score.

The study is organized into many sections that investigate the subject of " Comparative Analysis of CNN Architectures for English and Gujarati Speech Recognition Using MFCC Features" The text encompasses the overview of the study, focusing on CNN Architectures for English and Gujarati Speech Recognition. It provides a detailed literature review and outlines a methodology that employs comprehensive research approaches to analyze CNN architectures, with the objective of Speech Recognition Using MFCC Features. The text also presents the results and analysis derived from the study. Finally, it concludes by offering insights into potential future research directions pertaining to the issue.

REVIEW OF LITERATURE

The literature study includes the goals, methods, and findings of numerous respected researchers who studied the "Comparative Analysis of CNN Architectures for English and Gujarati Speech Recognition Using MFCC Features". Finding and evaluating publications that provide material relevant to the study topic is the exacting process of conducting a review of related literature. It describes the written part of a research plan or report that discusses the evaluated articles.

The work of [21] asserted that the ASR system uses the public Gujarati dataset. The study took an integrated front-end feature extraction strategy using Mel-frequency Cepstral Coefficients (MFCC) and Constant Q CQCC. MFCC and CQCC feature extraction approaches enhance Word Error Rate (WER) by 10–19% compared to isolated delta-delta features using the integrated model. Similarly the author [22] examined several Automatic Speech Recognition models, their applications in different fields, and lastly, it has assessed the accuracy of ASR for the most common Indian languages. The word done by [23] examined deep learning to recognize 10 Gujarati numbers from 0 to 9 (૦ to ૯). The total dataset has 8 native speakers, 4 male, and 4 female, aged 20–40. MFCC-CNN analyses audio samples to create spectrograms. The suggested model may be improved to examine speech recognition outcomes for relevant speaker factors including age, gender, and dialect. In a same way [24] Gammatone Cepstral Coefficients (GTCC) with Constant Q Cepstral Coefficients (CQCC) are combined in this work using integrated frontend feature extraction approaches. With the combined model, GTCC and CQCC feature extraction improves Word Error Rate (WER) by 9–16% over solo delta-delta features. The author [25] suggested a preliminary research employing mel-frequency cepstrum coefficients (MFCC) characteristics to detect human speech. The neural network results show that MFCC characteristics help identify speech. In a same way the study of [26] demonstrated that Voice signals from 367 speakers with 7 accents were utilized for MFCC feature extraction. Data for 330 speakers came from the UC Irvine Machine Learning (ML) open data source "Speaker Accent Recognition" data collection. So, ML algorithms' performance is displayed when the data set is partitioned into k pieces. The author [27] asserted that novel MFCC use for hand gesture recognition. MFCC for hand gesture recognition is used to test its image processing capabilities. Experimental findings show that MFCC gesture recognition may be utilized with other methods like Gabor filter and DWT to increase accuracy and efficiency. The work of [28] focused on developing speech recognition systems using shallow models, such as traditional ANNs and HMMs, together with Mel Frequency Cepstral Coefficients and other pertinent properties. The experimental findings show that the proposed CNN-based ASR system performs well for the Sylheti language, and the results are promising even if the system shows some training delay. The capacity of MFCC to extract characteristics that might be used to quantify the severity of speech impairment. Standardized telecommunications equipment and services that mitigate the negative effects of a handicap cannot be designed or produced without this technology [29]. The study of [30] suggested a variety of speech characteristics and ML models applicable to SER. It makes use of a deep learning algorithm to learn complicated multidimensional data and perform good categorization. Possible enhancements include improving the ML model and combining other features to get a higher true positive rate.

METHODOLOGY

The research methodology employs the Speech Commands dataset for English and a dataset of Gujarati digits for the investigation of speech recognition. Following pre-processing steps such as normalizing audio duration and using MFCC feature extraction, Convolutional Neural Networks (CNNs) are used for classification. The CNN designs exhibit variations in their depth, ranging from configurations with 2, 3, and 4 layers, all of which use ReLU activation functions. In addition, the study examines the Tanh activation function for the sake of comparison. The training process includes datasets in both English and Gujarati, to achieve resilience across different languages. Model performance and efficiency are balanced by varying key factors such as layer depth and optimization approaches. The experiment demonstrates the need for customized methods for voice detection in various language circumstances.

a. Dataset

English

The Speech Commands dataset is a specialized collection used in the development and assessment of speech recognition algorithms. It consists of audio files containing recordings of 30 words, including 20 core command words and 10 auxiliary words. This dataset is accessible through platforms like Dagshub and TensorFlow, providing researchers with valuable resources for their work in speech technology. These recordings include a range of vocal

pitches and accents, providing a diverse sample for robust model training. For effective training and testing of the machine learning models, the dataset was partitioned into three subsets. From the entire dataset, 51776 audio files were allocated for training. Training helps the model learn and adjust to different ways people speak commands. The study set aside 6,472 files for validation. This validation set is used to fine-tune the model's parameters and to evaluate its performance partway through the training process. Finally, another set of 6472 files was used as the test dataset. Using unseen data is essential for objectively evaluating the model's performance, ensuring it can handle new, unexposed data in real-world scenarios. This structured approach to splitting the dataset helps create a more reliable and accurate speech recognition system. Figure 1 shows the analysis of the dataset, including the number of files per word, and it reveals that the dataset is mostly balanced, but not completely.

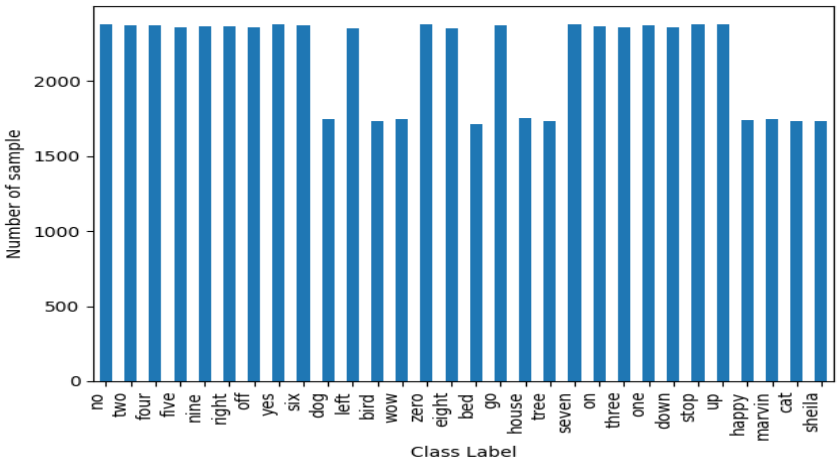


Figure 1: Class label with the number of instances per class

Source: Adopted from Y. Sharma [4]

Gujarati

The Gujarati dataset comprises authentic recordings of digits spoken by various users from five different regions of Gujarat[31]. It aims to support research on speech recognition systems by providing a simple audio dataset containing recordings of spoken digits in the WAV format at 44 kHz (Nyquist Frequency). The recordings were trimmed to minimize silence at the beginning and end, thereby capturing a variety of environmental conditions and background noise scenarios. The dataset consists of 1939 files, of which 1551 were allocated for training, 193 for validation, and 195 for testing purposes. Figure 2 illustrates the dataset analysis, which shows the number of files per word. This indicates a balanced dataset, which prevents bias, aids in accurate predictions and improves generalization.

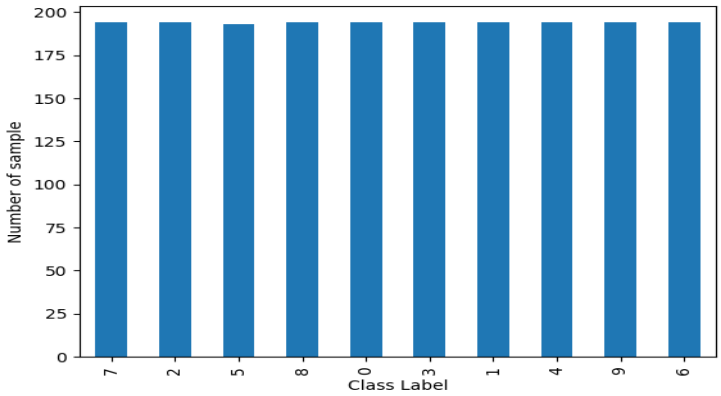


Figure 2: Class label with the number of instances per class.

Source: Adopted from Y. Sharma [4]

b. Pre-processing

Both datasets were initially pre-processed by adjusting the audio files to a uniform length of 16,000 samples through trimming or padding with zeros to ensure consistency for batch processing and model compatibility. Subsequently, the audio files were decoded into floating-point tensors and normalized to values between -1.0 and 1.0, promoting consistency and compatibility across different audio sources. Data augmentation techniques, including time shifting, adding background noise, and adjusting pitch and speed, were applied to diversify the dataset, thereby aiding the generalizability of the model. Spectrograms, which represent the frequency content of a waveform over time, were computed using Short-Time Fourier Transform (STFT), decomposing the audio signals into constituent frequencies over small, overlapping time windows. These spectrograms were then formatted into channels suitable for use with Convolutional Neural Networks (CNNs), ensuring compatibility with subsequent model architectures and facilitating the extraction of spatial features from the spectrogram images. These pre-processing steps collectively transform the raw audio data into a structured format conducive to subsequent feature extraction and model training.

c. Feature Extraction

Feature extraction involves converting a raw speech signal into a series of acoustic feature vectors that capture important information regarding speech. These features need to have specific characteristics: they should be robust against environmental noise, meaning that they remain reliable even in noisy conditions. Additionally, variations in voice due to factors such as the speaker's health or aging should not negatively affect the feature-extraction process. The features extracted from speech should be relatively easy to calculate and should be difficult to imitate or mimic using the speech of imposters. Moreover, it is important that the feature extraction process be computationally efficient and not easily replicable by imposters attempting to mimic speech. These attributes are essential for developing accurate and reliable speech recognition systems.

d. Mel Frequency Cepstral Coefficient (MFCC):

MFCC was used on both English and Gujarati datasets for feature extraction. MFCC is efficient for the feature extraction of speech signals because it mimics human hearing, provides a compact representation of the signal, is robust to noise and channel distortion, is time-invariant, and is easy to compute. These properties make the MFCC a powerful tool for a wide range of speech processing and recognition tasks. MFCCs are based on the known variation of the critical bandwidths of the human ear with frequency, and filters spaced linearly at low frequencies and logarithmically at high frequencies have been used to capture phonetically important characteristics of speech. This is expressed in the mel-frequency scale, which has a linear frequency spacing below 1000 Hz and a logarithmic spacing above 1000 Hz. Here, the Mel scale is being used which translates regular frequencies to a scale that is more appropriate for speech because the human ear perceives sound in a nonlinear manner.

The Mel scale for frequency f is determined using the equation:

$$Mel(f) = 2595\ log_{10}(\frac{f}{700} + 1) \qquad (1)$$

The logarithm of the magnitude of the Mel scale is referred to as the Mel spectrum. Then, the Discrete Cosine Transform (DCT) is applied to the Mel spectrum to compute the Mel frequency cepstral coefficients (MFCC features).

The following figure shows the steps involved in the MFCC feature extraction method.

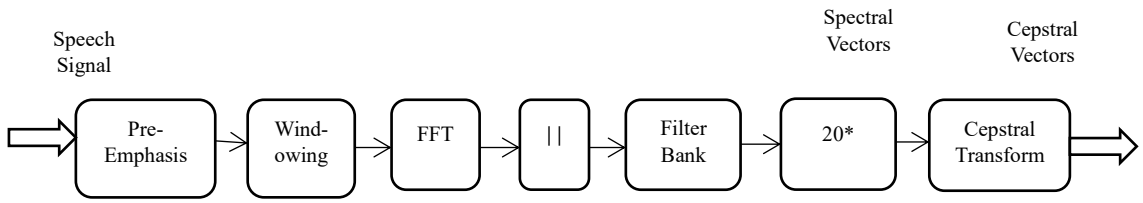


Figure 3: Block Diagram of MFCC [2].

MFCCs are derived in the following steps:

1) Pre-emphasis: In this step, a first-order high-pass filter is applied and then the speech signal is divided into small units of 20-25 milliseconds, which are called frames. This is achieved by applying a filter with a simple

transfer function:

$$H(z) = 1 - \alpha z^{-1} \quad (2)$$

Where α is a pre-defined co-efficient (commonly set to around 0.97)

2) Windowing: In this step window function generally, the hamming window is applied on each frame of the speech signal to reduce the discontinuities at the beginning and end of each frame. If the window being defined is Wn(m), 0 ≤ m ≤ Nm-1 where Nm stands for the number of samples within every frame, the output after windowing the signal will be presented as:

$$Y(m) = X(m)\, Wn(m), 0 \le m \le Nm - 1 \quad (3)$$

Where Y(m) represents the output signal after multiplying the input signal represented as x(m) and the Hamming window represented by Wn(m), which is usually represented as:

$$Wn(m) = 0.54 - 0.46 \cos\left(\frac{2\pi m}{Nm-1}\right), 0 \le m \le Nm - 1 \quad (4)$$

3) FFT: The Fast Fourier Transform converts each windowed frame from the time domain into the frequency domain. Fourier transformation is a fast algorithm to apply Discrete Fourier Transform (DFT), on the given set of Nm samples shown below:

$$D_k = \sum_{m=0}^{Nm-1} D_m e^{\frac{-j2\pi km}{Nm}} \quad (5)$$

Where k=0, 1, 2.... Nm-1

In FFT this frame will be divided into small DFTs and then the computation will be done on these divided small DFTs as individual sequences thus the computation will be faster and easier.

4) Mel Frequency Filter bank: The power spectrum obtained from the FFT is now mapped onto triangular mel scale filters.

5) LOG: Take LOG of the power at each of the mel frequencies.

6) DCT: In this step finally the log mel scale is converted back into the time domain by taking a discrete cosine transform of the log signal. The resultant parameter is called the mel frequency cepstral coefficients. The output after applying DCT is known as MFCC (Mel Frequency Cepstrum Coefficient)

$$C_n = \sum_{k-1}^{k} (\log D_k) \cos[\, m\left(k - \frac{1}{2}\right)\frac{\pi}{k}\,] \quad (6)$$

Where m = 0, 1... k- 1 where Cn represents the MFCC and m is the number of the coefficients.
The MFCCs we get can be used for different jobs in speech processing, like recognizing speech, identifying speakers, and understanding emotions.

e. Classifier

In the context of speech recognition, Convolutional Neural Networks (CNNs) are used to automatically learn discriminative features from spectrograms or other representations of speech signals. Typically, the CNN architecture comprises convolutional layers that are responsible for extracting spatial patterns or features from input speech data. These layers are followed by pooling layers, which reduce the dimensionality of the extracted features while retaining the crucial information. Through these layers, the network can capture hierarchical representations of speech data and learn high-level features from the low-level representations. Subsequently, additional fully connected layers and softmax layers are often incorporated for classification tasks, wherein the network predicts labels corresponding to the input speech samples. For speech recognition tasks involving both the English and Gujarati languages, CNNs can be trained on extensive datasets containing audio samples of words spoken in both languages. This enables the model to learn language-specific patterns and features, facilitating accurate recognition of spoken words in both English and Gujarati.

In this study, CNN architectures with 2-layer, 3-layer, and 4-layer configurations were considered. A 2-layer CNN typically consists of two convolutional layers, followed by pooling layers, offering a relatively simpler network structure. In contrast, a 3-layer CNN includes an additional convolutional layer, which enhances its ability to capture hierarchical features in data. A 4-layer CNN further deepens the network with an extra convolutional layer, potentially allowing for more intricate feature extraction and abstraction, albeit with increased computational

complexity and risk of overfitting.

f. Experimental Setup

Our speech recognition system relies on mel-frequency cepstral coefficients (MFCC) for feature extraction and Convolutional Neural Networks (CNN) for classification. This framework was applied to both English and Gujarati datasets. In this setup, CNN architectures with varying depths, including two-, three-, and 4-layer configurations, each employing a Rectified Linear Unit (ReLU) activation function, were implemented. Additionally, we experimented with the hyperbolic tangent (Tanh) activation function in a parallel setup with identical feature extraction (MFCC) and CNN classifier architecture. ReLU activation, defined as f(x) = max(0,x), returns an input value for positive inputs, and zero for negative inputs. In contrast, the Tanh function, with equation f(x) = e^x-/ e^x, produces smooth S-shaped outputs ranging between -1 and 1. Compared with ReLU, Tanh outputs tend to approach zero for small negative inputs, which can potentially mitigate noise amplification in speech signals. Our architectural decisions were guided by considerations of task complexity and computational resources, with the aim of achieving a balance between model performance and efficiency. Key hyperparameters, such as the number of layers, filters, dropout rates, and dense-layer units, vary across architectures. All models underwent training using the Adam optimizer and categorical cross-entropy loss function, representing a standard configuration for classification tasks.

The research methodology involves a methodical comparison of convolutional neural network (CNN) architectures for MFCC feature-based English and Gujarati voice recognition. This robust model training is made possible by the different linguistic samples offered by the selected datasets, which include Speech Commands for English and real recordings of digits for Gujarati. To improve the generalizability of the model, preprocessing is used to standardize audio duration and decode, normalize, and augment methods. Because of its effectiveness, resilience, and resemblance to human hearing, MFCC feature extraction is used. For the sake of comparison, CNN architectures with Tanh activation and those with ReLU activation are tested, spanning from 2-layer to 4-layer configurations. Categorical cross-entropy loss function and Adam optimizer training provide a consistent methodology across studies. This methodical approach makes it possible to compare CNN architectures in detail, which helps to understand their usefulness for voice recognition in various languages and promotes the creation of precise and successful speech recognition systems.

RESULTS

The comparative evaluation of the English and Gujarati datasets provides insights into the efficiency of our approach across different languages. By analyzing the performance metrics on both datasets, we gained a comprehensive understanding of the system's robustness and generalization capabilities across diverse linguistic contexts. Furthermore, we investigated any variations in the model behavior and performance between the two datasets, identifying potential challenges or opportunities specific to each language. Through this comparative analysis, we aim to provide valuable insights into the suitability and adaptability of our speech recognition system for multilingual applications, facilitating informed decision-making and enhancing the system's practical utility across different linguistic domains.

a. Training

The English dataset was trained for 50 epochs, while the Gujarati dataset underwent training for 100 epochs, with early stopping criteria implemented to prevent overfitting and ensure optimal model performance. Table 1 presents the training and validation accuracies of Convolutional Neural Network (CNN) architectures trained on both English and Gujarati datasets, utilizing ReLU and TanH activation functions. For the English dataset, across all architectures, models employing ReLU consistently outperformed those employing TanH activation in terms of both training and validation accuracies. In particular, the 3-layer CNN with ReLU activation demonstrated the highest validation accuracy of 0.90, demonstrating its efficacy in English speech recognition tasks. Conversely, for the Gujarati dataset, models with ReLU activation also exhibited superior performance compared with TanH across all architectures. However, the validation accuracies were generally lower for the Gujarati dataset than for the English dataset, suggesting potential challenges posed by linguistic nuances and dataset complexity. In summary, ReLU activation appears to be more effective than TanH for both English and Gujarati speech recognition tasks, and further optimization may be required to enhance the performance on the Gujarati dataset.

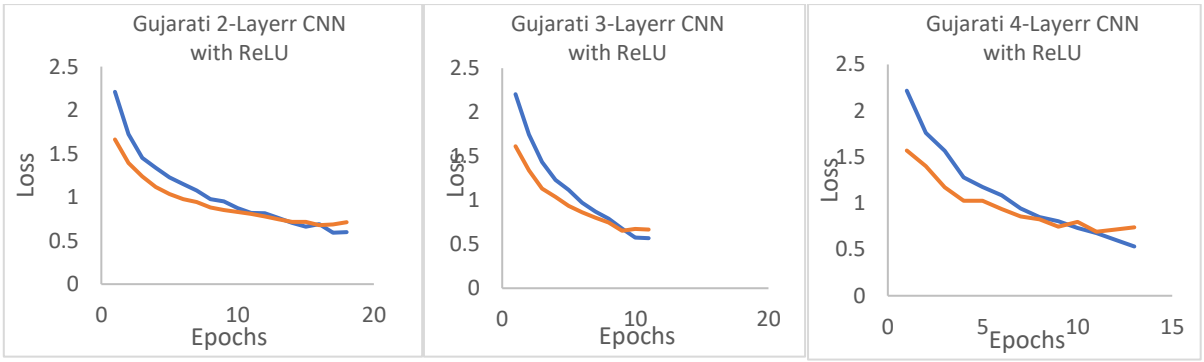
TABLE 1: Training logs for English datasets
Source: Authors own elaboration

CNN	English			
	ReLu		TanH	
	Training Accuracy	Validation Accuracy	Training Accuracy	Validation Accuracy
2-Layer	0.89	0.88	0.81	0.79
3-Layer	0.90	0.89	0.79	0.79
4-Layer	0.92	0.91	0.78	0.81

TABLE 2: Training logs for Gujarati datasets
Source: Authors own elaboration

CNN	Gujarati			
	RELu		Tanh	
	Training Accuracy	Validation Accuracy	Training Accuracy	Validation Accuracy
2-Layer	0.79	0.74	0.81	0.67
3-Layer	0.79	0.75	0.81	0.70
4-Layer	0.81	0.67	0.81	0.68

In the above table 2 and 3 the lower results observed with the Gujarati dataset could potentially be attributed to the smaller size and lower data diversity of the dataset. The Gujarati dataset comprised only 1939 audio files, with approximately 180 files per word, whereas the English dataset consisted of 64720 files, with approximately 2000 files per word. This significant difference in dataset size and data diversity may have limited the ability of the Gujarati models to generalize effectively, resulting in lower validation accuracies compared with the English dataset. Despite employing ReLU activation, which demonstrated superior performance across both datasets, the limited data availability in the Gujarati dataset may have hindered the capacity of the models to capture the intricacies of Gujarati speech patterns. Thus, future efforts should prioritize expanding and diversifying the Gujarati dataset to improve the model performance and generalization capabilities. The figure below shows graphs of training and validation loss over epochs for the English and Gujarati language datasets with ReLU and TanH activations. Figure 3 displays the training log graphs for the Gujarati dataset, whereas Figure 4 illustrates graphs for the English dataset with both activation functions. These graphs depict the reduction in loss with each epoch.



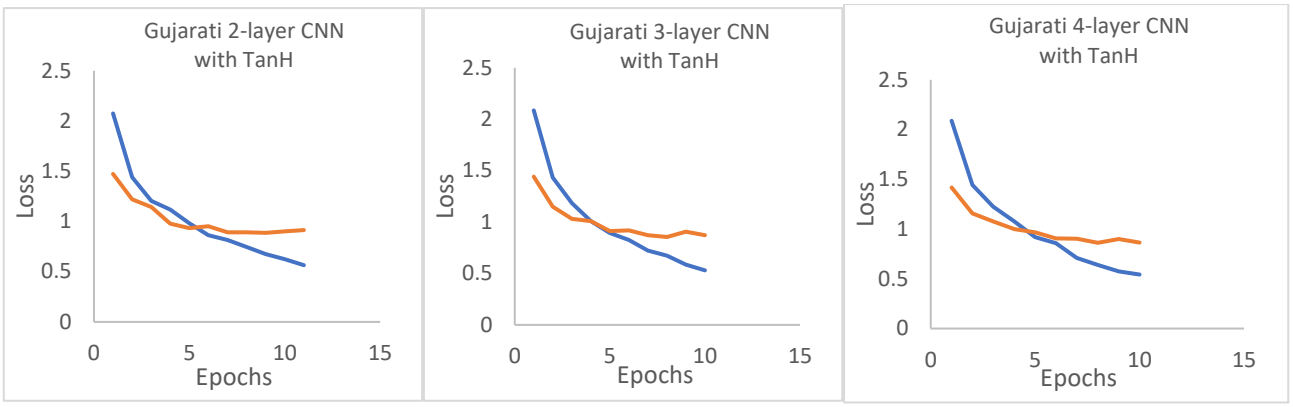


Figure 3: The training log graphs for the Gujarati dataset
Source: Authors own elaboration

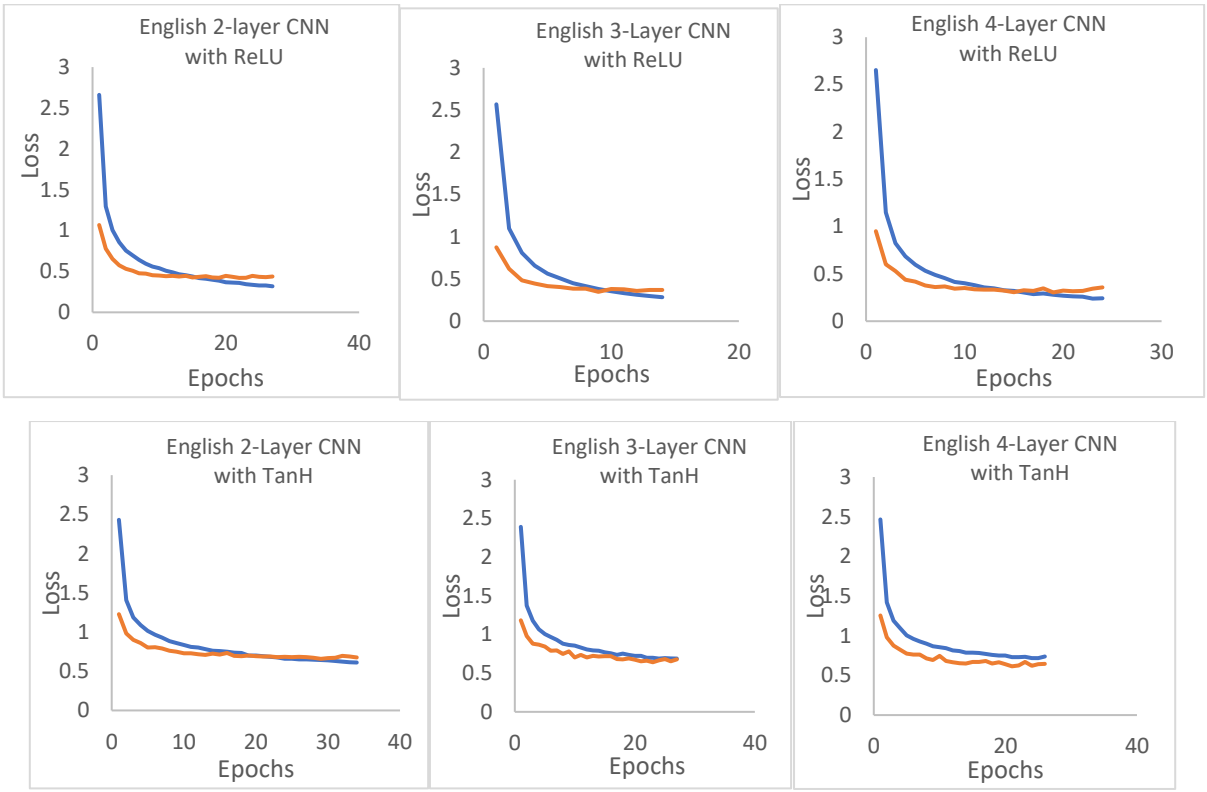


Figure 4: illustrates graphs for the English dataset
Source: Authors own elaboration

b. Testing

Testing was conducted on both datasets.

TABLE 3: Summary of the overall accuracy achieved by the model on the entire dataset

Source: Authors own elaboration

CNN	English		Gujarati	
	ReLU	TanH	ReLU	Tanh
2-Layer	0.88	0.79	0.73	0.72
3-Layer	0.90	0.80	0.75	0.69
4-Layer	0.91	0.81	0.72	0.69

Based on the results, a higher CNN architecture did not consistently translate to better performance across both datasets. Specifically, in the case of the Gujarati dataset, the 3-layer CNN outperformed its 4-layer counterpart, contradicting the expectation of increased accuracy with deeper networks. This observation suggests that the relationship between CNN depth and performance may vary depending on dataset characteristics and language complexities. Notably, the 4-layer CNN achieved the highest accuracy of 91% for English recognition, whereas in Gujarati, the 3-layer CNN attained the highest accuracy of 75%. Despite this inconsistency, across both datasets, the ReLU activation function consistently outperformed TanH, indicating its robustness in capturing the relevant features for speech recognition tasks. These findings underscore the importance of empirical evaluation and optimization of CNN architectures for specific language datasets to achieve optimal performance. Table 4 and 5 below presents detailed accuracy, precision, recall, and F1-score results class-wise for the English dataset.

TABLE 4: Detailed results of ReLU on the English dataset
Source: Authors own elaboration

Accuracy, P-Precision, R- Recall, F1- F1score

Words	**ReLU**											
	2-Layer				**3-Layer**				**4-Layer**			
	A	P	R	F1	A	P	R	F1	A	P	R	F1
no	0.87	0.83	0.87	0.85	0.9	0.88	0.9	0.89	0.9	0.87	0.9	0.88
two	0.86	0.84	0.86	0.85	0.89	0.9	0.89	0.89	0.91	0.9	0.91	0.9
four	0.93	0.84	0.93	0.88	0.91	0.94	0.91	0.92	0.93	0.93	0.93	0.93
five	0.85	0.89	0.85	0.87	0.87	0.9	0.87	0.88	0.85	0.96	0.85	0.9
nine	0.86	0.95	0.86	0.9	0.85	0.96	0.85	0.9	0.9	0.95	0.9	0.92
right	0.88	0.92	0.88	0.9	0.91	0.92	0.91	0.91	0.93	0.92	0.93	0.92
off	0.88	0.73	0.88	0.8	0.89	0.81	0.89	0.85	0.88	0.86	0.88	0.87
yes	0.94	0.94	0.94	0.94	0.92	0.96	0.92	0.94	0.95	0.99	0.95	0.97
six	0.91	0.94	0.91	0.93	0.93	0.93	0.93	0.93	0.91	0.97	0.91	0.94
dog	0.82	0.85	0.82	0.84	0.79	0.93	0.79	0.86	0.85	0.9	0.85	0.88
left	0.94	0.87	0.94	0.91	0.92	0.8	0.92	0.86	0.92	0.92	0.92	0.92
bird	0.9	0.92	0.9	0.91	0.89	0.92	0.89	0.91	0.9	0.92	0.9	0.91
wow	0.94	0.93	0.94	0.93	0.92	0.97	0.92	0.94	0.96	0.92	0.96	0.94
zero	0.91	0.94	0.91	0.93	0.92	0.94	0.92	0.93	0.92	0.97	0.92	0.95
eight	0.92	0.91	0.92	0.92	0.96	0.84	0.96	0.9	0.96	0.9	0.96	0.93
bed	0.87	0.89	0.87	0.88	0.87	0.92	0.87	0.89	0.94	0.9	0.94	0.92
go	0.79	0.74	0.79	0.77	0.81	0.84	0.81	0.82	0.88	0.82	0.88	0.85
house	0.88	0.97	0.88	0.92	0.94	0.95	0.94	0.95	0.9	0.99	0.9	0.95
tree	0.75	0.9	0.75	0.81	0.79	0.92	0.79	0.85	0.83	0.93	0.83	0.88
seven	0.92	0.94	0.92	0.93	0.94	0.96	0.94	0.95	0.94	0.95	0.94	0.95
on	0.9	0.83	0.9	0.86	0.92	0.91	0.92	0.92	0.93	0.86	0.93	0.89
three	0.85	0.76	0.85	0.8	0.85	0.79	0.85	0.82	0.88	0.83	0.88	0.85
one	0.92	0.93	0.92	0.93	0.92	0.94	0.92	0.93	0.96	0.87	0.96	0.91
down	0.8	0.89	0.8	0.84	0.87	0.92	0.87	0.89	0.87	0.89	0.87	0.88
stop	0.91	0.94	0.91	0.93	0.93	0.91	0.93	0.92	0.95	0.96	0.95	0.95
up	0.85	0.83	0.85	0.84	0.91	0.77	0.91	0.83	0.89	0.78	0.89	0.83
happy	0.86	0.96	0.86	0.91	0.97	0.89	0.97	0.93	0.96	0.96	0.96	0.96
Marvin	0.9	0.97	0.9	0.93	0.91	0.94	0.91	0.93	0.95	0.98	0.95	0.96
cat	0.89	0.82	0.89	0.85	0.94	0.92	0.94	0.93	0.9	0.93	0.9	0.91
sheila	0.93	0.93	0.93	0.93	0.94	0.95	0.94	0.95	0.95	0.95	0.95	0.95

The above table presents a comparison of the maximum and minimum accuracies for different English words across various CNN models and activation functions. Notably, the 2-layer ReLU model achieved the highest accuracy with the word "left," while "tree" recorded the lowest accuracy. Similarly, for the 3-layer ReLU model, "happy" reached the highest accuracy, whereas "tree" again had the lowest accuracy. The 4-layer ReLU model followed the same pattern, with "eight" achieving the highest accuracy and "tree" the lowest.

For models using the tanh activation function, the 2-layer tanh model showed the highest accuracy with the word "yes" and the lowest with "go," but "tree" had the second lowest accuracy. The 3-layer Tanh model achieved maximum accuracy with "zero" and minimum accuracy with "go." Finally, the 4-layer Tanh model recorded the highest accuracy with "happy" and the lowest with "tree," with "go" being the second lowest.

These results highlight a consistent trend: the word "tree" often shows lower accuracy, particularly with ReLU activation, likely due to its smaller representation in the training dataset and its phonetic similarity to the word "three." This suggests that both dataset size and phonetic similarity play significant roles in the performance of speech recognition models. Additionally, the variation in maximum accuracy words across different models and activation functions indicates that certain words are more easily recognized, depending on the specific architecture and activation function used. These insights underline the importance of considering both dataset composition and model configuration for optimizing speech recognition systems.

TABLE 5: Detailed results of each class on the English dataset Accuracy, P-Precision, R- Recall, F1- F1score

Source: Authors own elaboration

Words	**TanH**											
	2-Layer				**3-Layer**				**4-Layer**			
	A	P	R	F1	A	P	R	F1	A	P	R	F1
one	0.76	0.77	0.76	0.77	0.69	0.77	0.69	0.73	0.81	0.7	0.81	0.75
two	0.75	0.86	0.75	0.8	0.8	0.88	0.8	0.83	0.75	0.83	0.75	0.79
four	0.85	0.79	0.85	0.82	0.86	0.61	0.86	0.72	0.83	0.85	0.83	0.84
five	0.78	0.73	0.78	0.75	0.75	0.77	0.75	0.76	0.67	0.8	0.67	0.73
nine	0.77	0.82	0.77	0.79	0.77	0.84	0.77	0.8	0.8	0.88	0.8	0.84
right	0.87	0.76	0.87	0.81	0.83	0.84	0.83	0.84	0.87	0.67	0.87	0.76
off	0.8	0.66	0.8	0.72	0.74	0.76	0.74	0.75	0.74	0.8	0.74	0.77
yes	0.9	0.9	0.9	0.9	0.87	0.94	0.87	0.9	0.86	0.92	0.86	0.89
six	0.86	0.82	0.86	0.84	0.87	0.89	0.87	0.88	0.86	0.93	0.86	0.9
dog	0.68	0.79	0.68	0.73	0.72	0.76	0.72	0.74	0.71	0.76	0.71	0.74
left	0.86	0.82	0.86	0.84	0.85	0.85	0.85	0.85	0.87	0.79	0.87	0.83
bird	0.78	0.84	0.78	0.81	0.85	0.77	0.85	0.81	0.84	0.83	0.84	0.83
wow	0.85	0.75	0.85	0.79	0.84	0.85	0.84	0.84	0.9	0.84	0.9	0.87
zero	0.88	0.84	0.88	0.86	0.88	0.83	0.88	0.86	0.87	0.91	0.87	0.89
eight	0.86	0.88	0.86	0.87	0.85	0.88	0.85	0.86	0.88	0.8	0.88	0.84
bed	0.79	0.78	0.79	0.78	0.81	0.75	0.81	0.78	0.78	0.86	0.78	0.82
go	0.67	0.64	0.67	0.65	0.68	0.61	0.68	0.64	0.67	0.66	0.67	0.67
house	0.81	0.92	0.81	0.87	0.79	0.91	0.79	0.84	0.83	0.89	0.83	0.86
tree	0.69	0.76	0.69	0.72	0.72	0.76	0.72	0.74	0.64	0.94	0.64	0.76
seven	0.82	0.93	0.82	0.87	0.85	0.91	0.85	0.88	0.84	0.93	0.84	0.89
on	0.77	0.8	0.77	0.78	0.87	0.71	0.87	0.78	0.84	0.73	0.84	0.79
three	0.74	0.69	0.74	0.71	0.76	0.7	0.76	0.73	0.86	0.66	0.86	0.75
one	0.87	0.72	0.87	0.78	0.87	0.78	0.87	0.82	0.87	0.79	0.87	0.83
down	0.76	0.79	0.76	0.77	0.8	0.75	0.8	0.77	0.77	0.78	0.77	0.78
stop	0.83	0.84	0.83	0.84	0.76	0.87	0.76	0.81	0.76	0.92	0.76	0.84
up	0.72	0.79	0.72	0.76	0.74	0.83	0.74	0.78	0.8	0.71	0.8	0.75

happy	0.8	0.9	0.8	0.85	0.78	0.93	0.78	0.85	0.9	0.84	0.9	0.87
Marvin	0.74	0.87	0.74	0.8	0.82	0.86	0.82	0.84	0.83	0.86	0.83	0.84
cat	0.79	0.81	0.79	0.8	0.79	0.84	0.79	0.81	0.82	0.8	0.82	0.81
sheila	0.88	0.87	0.88	0.87	0.87	0.83	0.87	0.85	0.85	0.93	0.85	0.89

Table 6 and 7 below presents detailed accuracy, precision, recall, and F1-score results class-wise for the Gujarati dataset.

TABLE 6: Detailed results ReLU on the Gujarati dataset

Source: Authors own elaboration

Words	**ReLU**											
	2-Layer				**3-Layer**				**4-Layer**			
	A	P	R	F1	A	P	R	F1	A	P	R	F1
0	0.83	0.88	0.83	0.86	0.83	0.83	0.83	0.83	0.78	0.93	0.78	0.85
1	0.68	0.77	0.68	0.72	0.84	0.78	0.84	0.81	0.72	0.75	0.72	0.73
2	0.64	0.67	0.64	0.65	0.68	0.65	0.68	0.67	0.73	0.70	0.73	0.71
3	0.67	0.38	0.67	0.49	0.73	0.41	0.73	0.52	0.73	0.48	0.73	0.58
4	0.83	0.95	0.83	0.88	0.78	0.86	0.78	0.82	0.87	0.87	0.87	0.87
5	0.75	0.83	0.75	0.79	0.70	0.93	0.70	0.80	0.60	0.80	0.60	0.69
6	0.94	0.71	0.94	0.81	0.89	0.84	0.89	0.86	0.89	0.70	0.89	0.78
7	0.65	0.85	0.65	0.73	0.71	0.80	0.71	0.75	0.76	0.72	0.76	0.74
8	0.55	0.79	0.55	0.65	0.60	0.92	0.60	0.73	0.40	0.80	0.40	0.53
9	0.82	0.70	0.82	0.76	0.76	0.76	0.76	0.76	0.82	0.67	0.82	0.74

TABLE 7: Detailed results of TanH on the Gujarati dataset

Source: Authors own elaboration

Words	**TanH**											
	2-Layer				**3-Layer**				**4-Layer**			
	A	P	R	F1	A	P	R	F1	A	P	R	F1
0	0.78	0.93	0.78	0.85	0.83	0.94	0.83	0.88	0.78	0.78	0.78	0.78
1	0.72	0.75	0.72	0.73	0.76	0.79	0.76	0.78	0.84	0.72	0.84	0.78
2	0.73	0.70	0.73	0.71	0.59	0.76	0.59	0.67	0.59	0.76	0.59	0.67
3	0.73	0.48	0.73	0.58	0.73	0.41	0.73	0.52	0.87	0.45	0.87	0.59
4	0.87	0.87	0.87	0.87	0.78	0.90	0.78	0.84	0.78	0.75	0.78	0.77
5	0.60	0.80	0.60	0.69	0.45	0.90	0.45	0.60	0.50	0.83	0.50	0.63
6	0.89	0.70	0.89	0.78	0.78	0.74	0.78	0.76	0.78	0.74	0.78	0.76
7	0.76	0.72	0.76	0.74	0.71	0.60	0.71	0.65	0.71	0.75	0.71	0.73
8	0.40	0.80	0.40	0.53	0.60	0.67	0.60	0.63	0.45	0.90	0.45	0.60
9	0.82	0.67	0.82	0.74	0.76	0.54	0.76	0.63	0.71	0.57	0.71	0.63

The table above details the maximum and minimum accuracies for different Gujarati digits across the various CNN models and activation functions. For models using the ReLU activation function, the 2-layer, 3-layer, and 4-layer CNNs all achieved their highest accuracy with the digit "6" and their lowest accuracy with the digit "8." This consistency indicates that the digit "6" is more easily recognized across different depths of CNNs using ReLU, while "8" consistently presents more difficulty.

For models utilizing the tanh activation function, the 2-layer tanh model also achieved the highest accuracy with the digit "6" and the lowest with "8," aligned with the performance observed in the ReLU models. However, the

3-layer tanh model showed a unique pattern, achieving the highest accuracy with the digit "0" and the lowest with "5." The 4-layer tanh model had the highest accuracy with the digit "3" and the lowest with "8."

These findings provide several insights. First, the digit "6" appears to be consistently recognized with high accuracy across various models, indicating that it has distinct features that are well captured by both the ReLU and tanh activation functions. Conversely, the digit "8" is frequently associated with the lowest accuracy, likely due to its phonetic similarity to the digit "7" ("Saat" Saat'), making it challenging for the models to distinguish between them. Additionally, the variation in the highest accuracy digits for tanh models suggests that different configurations of CNNs and activation functions can significantly impact the model performance, emphasizing the need for tailored approaches depending on the specific dataset and model architecture. These insights are crucial for optimizing speech recognition systems for Gujarati digits, considering both the strengths and weaknesses observed in the model performance. The graph below displays a comparison between the accuracy, precision, F1-score, and recall of ReLU and TanH on three architectures for the Gujarati dataset.

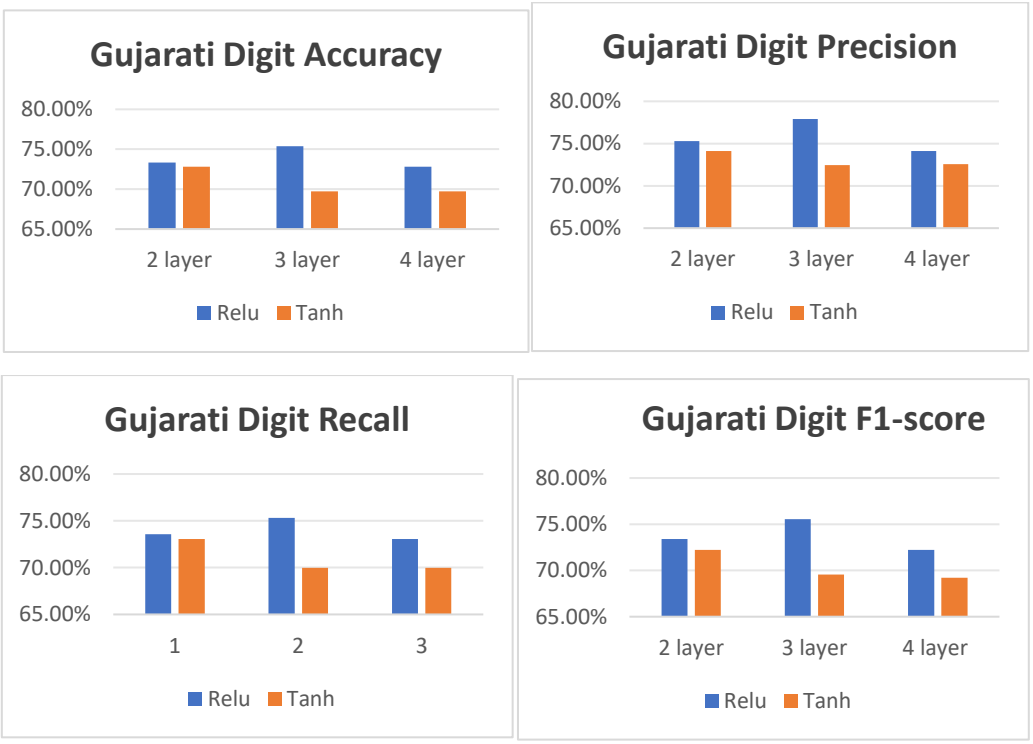


Figure 5: A comparison between the accuracy, precision, F1-score, and recall of ReLU and TanH on three architectures for the Gujarati dataset.

Source: Authors own elaboration

The results obtained on the Gujarati dataset showed varying performance metrics across different CNN architectures and activation functions. For models utilizing ReLU activation, the 3-layer CNN exhibited the highest accuracy of 75.38%, while the 4-layer CNN achieved the highest precision of 74.11%. However, models with Tanh activation generally yielded lower performance metrics than ReLU, with the 2-layer CNN recording the highest accuracy of 72.82%. Notably, the recall and F1 score metrics also followed a similar trend, with ReLU consistently outperforming Tanh across all the architectures. The graph below displays a comparison between the accuracy, precision, F1-score, and recall of ReLU and TanH on three architectures on the English dataset.

In contrast to the Gujarati dataset, the English dataset displayed higher performance metrics across both ReLU and Tanh activation functions. Among the different CNN architectures, the 4-layer CNN with ReLU activation demonstrated the highest accuracy of 91.29%, while the 4-layer CNN with Tanh activation achieved the highest precision of 82.15%. Similarly, the ReLU activation consistently outperformed Tanh, with higher accuracy, precision, recall, and F1 score metrics across all architectures.

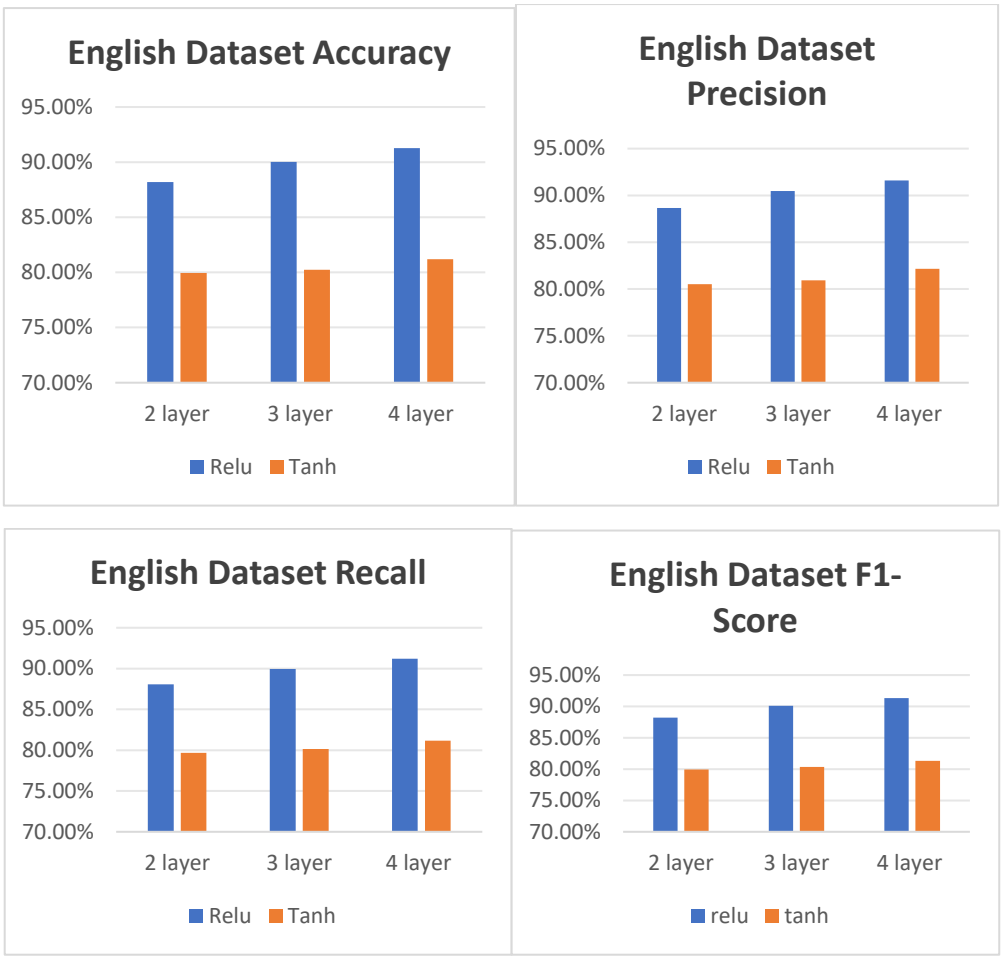


Figure 6: A comparison between the accuracy, precision, F1-score, and recall of ReLU and TanH on three architectures on the English dataset.

Source: Authors own elaboration

In contrast to the Gujarati dataset, the English dataset displayed higher performance metrics across both ReLU and Tanh activation functions. Among the different CNN architectures, the 4-layer CNN with ReLU activation demonstrated the highest accuracy of 91.29%, while the 4-layer CNN with Tanh activation achieved the highest precision of 82.15%. Similarly, the ReLU activation consistently outperformed Tanh, with higher accuracy, precision, recall, and F1 score metrics across all architectures.

CONCLUSION

A comparison of the results between the Gujarati and English datasets reveals notable disparities in model performance. While ReLU activation generally yielded superior performance across both datasets, the English dataset consistently displayed higher accuracy and other performance metrics than Gujarati. This discrepancy can be attributed to the smaller size and lower diversity of the Gujarati dataset, which likely limited the ability of the models to generalize effectively. Moreover, the linguistic nuances and dataset complexity inherent in the Gujarati language may pose additional challenges to accurate speech recognition. In CNN architectures, ReLU activation routinely performed better than TanH in the comparison of the English and Gujarati datasets. With ReLU activation in the 4-layer CNN, the maximum accuracy obtained in English was 91.29%; in Gujarati, the best accuracy was 75.38% with the 3-layer ReLU CNN. Because of its smaller quantity and less varied data, the Gujarati dataset did not do as well overall as English. Remarkably, several words, such as "tree" in English and "8" in Gujarati, continuously displayed less accuracy. Higher accuracy, precision, recall, and F1 score metrics were consistently produced by ReLU than by TanH across all architectures.

DISCUSSIONS

The study revealed an intriguing observation regarding the relationship between CNN architecture and performance. Contrary to the common belief that deeper CNNs inherently yield better results, the study found that this may not hold true for all datasets and languages. Specifically, in the case of the Gujarati dataset, the 3-layer CNN consistently outperformed the 4-layer CNN, irrespective of the activation function employed. This suggests that the complexity introduced by additional layers may not always translate into improved performance, particularly in scenarios with limited data or linguistic intricacies.

However, findings also highlight a contrasting pattern in the English dataset, where the 4-layer CNN demonstrated superior performance, particularly with ReLU activation. This disparity underscores the importance of considering dataset characteristics and linguistic nuances when designing CNN architectures for speech recognition tasks. Moreover, the consistent superiority of ReLU activation over TanH activation across both datasets emphasizes the critical role of activation functions in shaping model performance.

Additionally, the study underscores the need for further research to explore the interplay between CNN architecture, dataset characteristics, and language-specific features in speech recognition tasks. By gaining deeper insight into these factors, researchers can develop more effective and robust CNN models tailored to diverse linguistic contexts. Overall, our findings provide valuable insights into the optimization of CNNs for speech recognition applications, offering guidance for future research endeavors in this domain.

LIMITATIONS

i. Few Gujarati databases prevent in-depth investigation and can make results less applicable.

ii. Recognizing whole words or phrases in Gujarati is more difficult than focusing in only on the numbers.

iii. It is possible that the complexities of different language contexts will be too great for MFCC features and CNN structures to handle.

iv. It is essential to look at other approaches than MFCC and CNN, to make speech recognition systems more flexible and resilient.

REFERENCES

[1] VimalaC, “A Review on Speech Recognition Challenges and Approaches.”

[2] A. M. Sharma, “Speaker Recognition Using Machine Learning Techniques,” San Jose State University, San Jose, CA, USA, 2019. doi: 10.31979/etd.fhhr-49pm.

[3] D. Honnavalli and S. S. Shylaja, “Supervised Machine Learning Model for Accent Recognition in English Speech Using Sequential MFCC Features,” 2021, pp. 55–66. doi: 10.1007/978-981-15-3514-7_5.

[4] Y. Sharma, B. Abraham, and P. Jyothi, “Gujarati-English Code-Switching Speech Recognition using ensemble prediction of spoken language,” Mar. 2024, [Online]. Available: http://arxiv.org/abs/2403.08011

[5] A. S. Dhanjal and W. Singh, “A comprehensive survey on automatic speech recognition using neural networks,” *Multimed Tools Appl*, Aug. 2023, doi: 10.1007/s11042-023-16438-y.

[6] D. Raval, V. Pathak, M. Patel, and B. Bhatt, “Improving Deep Learning based Automatic Speech Recognition for Gujarati,” *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 21, no. 3, pp. 1–18, May 2022, doi: 10.1145/3483446.

[7] C. Ittichaichareon, S. Suksri, and T. Yingthawornsuk, “Speech recognition using MFCC,” in *International conference on computer graphics, simulation and modeling*, 2012.

[8] S. Mendiratta, N. Turk, and D. Bansal, “A Robust Isolated Automatic Speech Recognition System using Machine Learning Techniques,” *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 10, pp. 2325–2331, Aug. 2019, doi: 10.35940/ijitee.J8765.0881019.

[9] J. H. Tailor and D. B. Shah, “HMM-Based Lightweight Speech Recognition System for Gujarati Language,” 2018, pp. 451–461. doi: 10.1007/978-981-10-3920-1_46.

[10] S. Valaki and H. Jethva, “A hybrid HMM/ANN approach for automatic Gujarati speech recognition,” in *2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, IEEE, Mar. 2017, pp. 1–5. doi: 10.1109/ICIIECS.2017.8276141.

[11] Z. Ali, A. W. Abbas, T. M. Thasleema, B. Uddin, T. Raaz, and S. A. R. Abid, "Database development and automatic speech recognition of isolated Pashto spoken digits using MFCC and K-NN," *Int J Speech Technol*, vol. 18, no. 2, pp. 271–275, Jun. 2015, doi: 10.1007/s10772-014-9267-z.

[12] A. A. Ayranci, S. Atay, and T. Yildirim, "Speaker Accent Recognition Using Machine Learning Algorithms," in *2020 Innovations in Intelligent Systems and Applications Conference (ASYU)*, IEEE, Oct. 2020, pp. 1–6. doi: 10.1109/ASYU50717.2020.9259902.

[13] P. Pandit, S. Bhatt, and P. Makwana, "Automatic speech recognition of Gujarati digits using artificial neural network," in *Proceedings of 19th Annual Cum 4th International Conference of GAMS On Advances in Mathematical Modelling to Real World Problems*, 2014, pp. 141–146.

[14] N. Aasofwala, S. Verma, and K. Patel, "NLP based model to convert English speech to Gujarati text for deaf & dumb people," in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, IEEE, Jul. 2023, pp. 1–6. doi: 10.1109/ICCCNT56998.2023.10308284.

[15] Z. Song, "English speech recognition based on deep learning with multiple features," *Computing*, vol. 102, no. 3, pp. 663–682, Mar. 2020, doi: 10.1007/s00607-019-00753-0.

[16] K. Naithani, V. M. Thakkar, and A. Semwal, "English Language Speech Recognition Using MFCC and HMM," in *2018 International Conference on Research in Intelligent and Computing in Engineering (RICE)*, IEEE, Aug. 2018, pp. 1–7. doi: 10.1109/RICE.2018.8509046.

[17] B. Bhagat and M. Dua, "Enhancing Performance of Noise-Robust Gujarati Language ASR Utilizing the Hybrid Acoustic Model and Combined MFCC + GTCC Feature," 2024, pp. 221–231. doi: 10.1007/978-981-99-8129-8_19.

[18] P. Pandit and S. Bhatt, "Automatic speech recognition of Gujarati digits using wavelet coefficients in machine learning algorithms," *International Journal of Innovative Computing and Applications*, vol. 14, no. 4, pp. 191–200, 2023, doi: 10.1504/IJICA.2023.134184.

[19] A.-L. Rusnac and O. Grigore, "CNN Architectures and Feature Extraction Methods for EEG Imaginary Speech Recognition," *Sensors*, vol. 22, no. 13, p. 4679, Jun. 2022, doi: 10.3390/s22134679.

[20] O. Abdel-Hamid, A. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional Neural Networks for Speech Recognition," *IEEE/ACM Trans Audio Speech Lang Process*, vol. 22, no. 10, pp. 1533–1545, Oct. 2014, doi: 10.1109/TASLP.2014.2339736.

[21] M. Dua and Akanksha, "Gujarati Language Automatic Speech Recognition Using Integrated Feature Extraction and Hybrid Acoustic Model," 2023, pp. 45–54. doi: 10.1007/978-981-19-7753-4_4.

[22] A. Jha, P. K. Kushwaha, A. P. Srivastava, A. Thakur, D. Kumar, and S. Gupta, "Enabling Speech Recognition for Lesser-Known Language," in *2023 6th International Conference on Contemporary Computing and Informatics (IC3I)*, IEEE, Sep. 2023, pp. 348–353. doi: 10.1109/IC3I59117.2023.10397667.

[23] 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, pp. 424–429, 2022, doi: 10.14569/IJACSA.2022.0130450.

[24] A. Akanksha, "Tamil Language Automatic Speech Recognition Based on Integrated Feature Extraction and Hybrid Deep Learning Model," 2023, pp. 283–292. doi: 10.1007/978-981-19-9719-8_23.

[25] P. Barua, K. Ahmad, A. A. S. Khan, and M. Sanaullah, "Neural network based recognition of speech using MFCC features," in *2014 International Conference on Informatics, Electronics & Vision (ICIEV)*, IEEE, May 2014, pp. 1–6. doi: 10.1109/ICIEV.2014.6850680.

[26] A. A. AYRANCI, S. ATAY, and T. YILDIRIM, "Speaker Accent Recognition Using MFCC Feature Extraction and Machine Learning Algorithms," *International Journal of Advances in Engineering and Pure Sciences*, vol. 33, pp. 17–27, Dec. 2021, doi: 10.7240/jeps.896427.

[27] S. Gupta, J. Jaafar, W. F. wan Ahmad, and A. Bansal, "Feature Extraction Using Mfcc," *Signal Image Process*, vol. 4, no. 4, pp. 101–108, Aug. 2013, doi: 10.5121/sipij.2013.4408.

[28] G. Chakraborty, M. Sharma, N. Saikia, and K. K. Sarma, "Soft-computation based speech recognition system for Sylheti language," *Int J Speech Technol*, vol. 25, no. 2, pp. 499–509, Jun. 2022, doi: 10.1007/s10772-022-09976-7.

[29] G. Jhawar, P. Nagraj, and P. Mahalakshmi, "Speech disorder recognition using MFCC," in *2016 International Conference on Communication and Signal Processing (ICCSP)*, IEEE, Apr. 2016, pp. 0246–0250. doi: 10.1109/ICCSP.2016.7754132.

[30] H. S. Kumbhar and S. U. Bhandari, "Speech Emotion Recognition using MFCC features and LSTM network," in *2019 5th International Conference On Computing, Communication, Control And Automation (ICCUBEA)*, IEEE, Sep. 2019, pp. 1–3. doi: 10.1109/ICCUBEA47591.2019.9129067.

[31] N. Dalsaniya, S. H. Mankad, S. Garg, and D. Shrivastava, "Development of a Novel Database in Gujarati Language for Spoken Digits Classification," 2020, pp. 208–219. doi: 10.1007/978-981-15-4828-4_18.

ACKNOWLEDGEMENT

I like to convey my deep appreciation to all those who have provided support and served as a source of inspiration throughout this endeavor. I express my gratitude to my mentors for their valuable counsel, my coworkers for their effective teamwork, and my friends for their unwavering support. I am appreciative of the institutions and organizations that have provided resources and opportunity. Finally, I express my sincere gratitude to my family for their steadfast affection and assistance. The attainment of this accomplishment would not have been feasible without the combined participation of every one among you.