Chapter 6

Computational Experiments

6. Introduction

Recent developments in speech recognition, natural language processing, and voice-activated technology have all been made possible by the advancement of computational techniques, which have fundamentally changed many facets of human communication. The computational aspect of stammering, particularly in the context of the Hindi language, has received relatively little attention. The importance of computational research in treating stammering is examined in this chapter, which also emphasizes the necessity of creating computational methods specifically for Hindi speakers who stammer. Hindi-stammering people may gain from voice assistants and other voice-controlled technologies by utilizing computational methods, improving their communication experiences, and easing their integration into the digital world.

Stammering affects people of all linguistic and cultural backgrounds, making effective communication difficult and having an adverse effect on social interactions. While traditional speech therapies and therapeutic interventions are essential for managing stammering, the incorporation of computational techniques opens new avenues for helping people who stammer, particularly in the context of technology-driven communication.

Stammering computational research holds great promise for creating novel solutions that can help people with their daily communication needs. In our digital lives, voice assistants like Siri, Alexa, and Google Assistant are commonplace. However, they frequently have a limited impact on people who stammer, particularly when speaking languages like Hindi. The usability and accessibility of voice assistants for this particular population can be greatly improved by the development of reliable and accurate speech recognition models that are adapted to the particular speech patterns and difficulties faced by people who stammer.

Additionally, improvements in machine learning and natural language processing techniques offer chances to create individualized interventions and support tools for people who stammer. It is now possible to analyze speech patterns, spot disfluencies, and offer real-time feedback and interventions that support better communication and fluency by utilizing the power of computational algorithms. By enabling them to interact more effectively and confidently with voice-controlled technologies, these computational tools can empower people who stammer and provide them with new opportunities for participation in the digital world.

Despite the growth of computational stammering research, there is still a significant gap in services for Hindi-speaking stammerers. One of the most widely used languages in India, Hindi has a significant cultural and linguistic impact. However, there has not been much progress in the creation of computational methods and software programs that are tailored specifically for Hindi stammering. We can help close this gap and make it easier for people who stammer to use voice-controlled technologies that support their native language by highlighting the value of computational research in the context of Hindi stammering.

This chapter will look at computational methods and research that are currently available in the field of stammering, with a particular emphasis on the importance of using these methods to treat

Hindi stammering. We can pave the way for the creation of innovative and inclusive technologies that improve the communication experiences of this particular population by understanding the special difficulties faced by Hindi speakers who stammer and the potential advantages of computational interventions.

6.1 Machine learning

Machine learning uses a variety of methods for binary classification tasks, especially when working with images (Vieira et al., 2019). Here are some methods that are frequently used:

- **Logistic Regression**

  A well-liked method for binary classification is logistic regression. A logistic function is used to model the likelihood that an instance belongs to a particular class. Logistic regression can be used to classify images by using features that were extracted from the images as inputs (Connelly, 2020).

- **Support Vector Machines (SVM)**

  SVM is a potent supervised learning algorithm that uses a hyperplane to classify data points into various groups. SVM can be combined with different kernel functions in image classification to learn a decision boundary between the two classes (Gandhi, 2018).

- **Random Forests**

  As an ensemble learning technique, random forests combine various decision trees to produce predictions. The final prediction is established by the majority vote of the individual decision trees, each of which is constructed using a subset of the data. By using image features as input to the decision trees, random forests can be effective for image classification tasks (Biau & Scornet, 2016).

- **Convolutional Neural Networks (CNN)**

CNNs are incredibly effective at classifying images. By utilizing convolutional layers, pooling layers, and fully connected layers, they are specifically created to capture spatial hierarchies and patterns in images. CNNs are particularly suited for image classification because they automatically identify pertinent features from the data (Saxena, 2022).

- **Deep Learning Architectures**

  Besides CNNs, other deep learning architectures such as recurrent neural networks (RNNs) and long short-term memory (LSTM) networks can be employed for binary image classification. When working with images-related sequential or time-series data, these architectures are especially helpful.

- **Gradient boosting**

  Gradient boosting is an ensemble learning technique that sequentially combines weak classifiers, with each new classifier being trained to fix the errors of the previous ones. Many binary classification tasks, including image classification, have shown excellent performance from gradient boosting algorithms like XGBoost and LightGBM (Natekin & Knoll, 2013).

- **Transfer Learning**

  In this technique, a pre-trained model—typically one that has been trained on a sizable dataset—is used as the foundation for a brand-new classification task. Even with little training data, transfer learning can significantly enhance the performance of a binary image classifier by utilizing the learned features from a pre-trained model (Iman et al., 2023).

6.2 Classification

To delve deeper into this subject, we first introduce a common method for audio classification that involves spectrogram transformation of audio signals and treating them as images for classification (Liao et al., 2015).

- **Image based classification**

  The ability to use image-based classification techniques on audio data opens a world of possibilities when audio signals are transformed into spectrograms. A visual representation of the frequencies present in an audio signal over time, referred to as a spectrogram, is given. By utilizing this transformation, we can efficiently analyze and classify audio samples using a variety of image classification techniques, including deep learning models like convolutional neural networks (CNNs). We will examine the machine learning methods used for binary classification tasks, concentrating on audio data. Our goal is to gain a thorough understanding of the procedure for spectrogramming audio signals and applying image classification methods to audio classification. Our goal is to improve our capacity to correctly classify and categorize audio samples to enable a variety of applications, including speech recognition, audio event detection, and music genre classification. We will delve into the fundamental ideas and classification methodologies through this investigation. We will investigate a variety of machine learning methods, such as logistic regression, support vector machines (SVM), random forests, and the potent field of deep learning architectures. We will also talk about the use of pre-trained models and the potential of transfer learning for audio classification tasks. We want to improve our capacity for accurately classifying audio samples and unlock the potential for new applications and developments in the field of audio analysis by exposing the subtleties of audio classification techniques (Jiang, 2021).

6.3 Audio to Spectrogram Conversion

The conversion of an audio signal into a spectrogram shows the frequencies that are present in an audio signal over time visually (Almeida, 1994). The following actions are typically taken to transform an audio signal into a spectrogram:

- **Preprocessing**

  To enhance the quality of the data, the audio signal may go through preprocessing procedures like resampling, normalization, or noise removal.

- **Short-time Fourier Transform (STFT)**

  To obtain the frequency representation, the Fourier Transform is applied to each of the audio signal's brief, overlapping frames.

- **The power spectrum Calculation**

  The magnitude of the Fourier Transform is squared to obtain the power spectrum, representing the energy distribution across different frequencies.

- **Windowing**

  To minimize spectral leakage and produce more aesthetically pleasing spectrogram representations, window functions (such as the Hamming window) are applied to each frame.

- **Logarithmic scaling**

  To highlight important characteristics and condense the dynamic range, the power spectrum values are frequently converted to a logarithmic scale.

6.4 Spectrogram as Image

The spectrogram that is created from the audio signal can then be treated as a 2D image, with the x-axis standing for time, the y-axis for frequency, and the color or intensity standing for the strength or power of the corresponding frequency component.

- **Image Classification Techniques**

  When the spectrogram is represented as an image, image classification methods can be used to divide audio into various groups. Typical strategies include:

- **Convolutional Neural Networks (CNNs)**

  Strong deep learning models known as CNNs are frequently used for image classification tasks. Using the spatial hierarchies and patterns in the spectrogram, they can be directly applied to the spectrogram images, much like how they are used for image classification tasks.

- **Transfer Learning**

  For audio classification tasks, pre-trained CNN models, such as those trained on sizable image datasets like ImageNet, can be fine-tuned on the spectrogram images. This strategy works especially well when the audio dataset is small because it makes use of the expertise gained from large-scale image datasets.

- **Traditional Image Classification Techniques**

  The spectrogram images can also be subjected to conventional image classification algorithms, such as SVM, random forests, or logistic regression, by treating them as regular images with features taken from the spectrogram as input.

6.5 Audio classification

There are methods for categorizing audio without directly transforming the data into spectrograms. These methods, which operate directly on the raw audio signals, are frequently

referred to as "raw audio" or "waveform-based" approaches (Nanni et al., 2021). Here are a few illustrations:

- **Waveform-based Convolutional Neural Networks (CNNs)**

  CNNs can be used to analyze the raw audio waveforms without the use of spectrograms. The CNN architecture is capable of learning to directly extract pertinent features from the waveform data and can be designed to accept 1D input signals. This method can simultaneously capture both temporal and frequency information and does away with the need for explicit spectrogram conversion.

- **WaveNet**

  A deep generative model called WaveNet was created specifically to work with unprocessed audio waveforms. It makes use of dilated convolutions, allowing the receptive field to expand exponentially without significantly increasing the number of parameters. Both generative tasks (like audio synthesis) and discriminative tasks (like audio classification) can be performed using WaveNet models.

- **Long Short-Term Memory (LSTM) Networks**

  Recurrent neural networks of a certain kind, called LSTMs, can simulate long-term dependencies in sequential data. They can learn to recognize temporal patterns and dynamics in the audio data by directly processing the raw audio waveforms with LSTMs. In audio classification tasks, LSTM-based models have demonstrated success, especially when working with time-series audio data.

- **Raw Audio with Traditional Machine Learning Techniques**

  Raw audio waveforms can also be used directly by conventional machine learning algorithms like SVM, random forests, or logistic regression. MFCC (Mel Frequency

Cepstral Coefficients) or filter bank energies, for example, can be extracted from the audio signals and used as input features for the classifiers in this situation.

- **Hybrid Approaches**

  Additionally, it is possible to combine spectrogram-based and raw audio techniques. For instance, a model may have parallel branches, with one branch using CNNs or LSTMs to process the raw audio waveforms and the other using CNNs to analyze the spectrogram representation. To reach the ultimate classification conclusion, the outputs from both branches can be combined and fed into a fusion model.

6.6 Pros and cons analysis

There are two main methods that are frequently used in audio classification: spectrogram-based and raw audio waveform-based methods. Each strategy has its own set of advantages and disadvantages, making it appropriate for various situations. The spectrogram-based approach entails the transformation of audio signals into spectrograms, which are visual representations that track the frequency content of the audio over time. The raw audio waveform-based approach, on the other hand, works directly with unaltered audio signals. We will now examine these two methods, their importance in audio classification, and how both have unique benefits for classifying and analyzing audio data.

Pros of Spectrogram-Based Approach

- **Frequency-Time Representation**

Spectrograms give an audio signal a frequency-time representation that enables simultaneous capture of spectral and temporal data. Tasks requiring the analysis of frequency content and changes over time may benefit from this.

- **Audio to Image Domain Translation**

  By transforming audio into an image representation, spectrogram-based methods can make use of potent image processing and computer vision techniques, including deep learning models like CNNs, which have excelled in image classification tasks.

- **Visualization and Interpretability**

  Spectrograms provide an audio data representation that is visually comprehensible. The spectrogram's color or intensity patterns can reveal information about the predominate frequencies and time-varying properties of the audio signals.

Cons of Spectrogram-Based Approach

- **Loss of Phase Information**

  Since spectrograms only depict the magnitude or power spectrum of the audio signals, they omit phase information. This phase information loss may make it more difficult to accurately reconstruct the original audio waveform, which may be essential for some audio tasks like voice recognition or audio synthesis.

- **Information compression**

  Spectrograms condense audio data into a representation as a 2D image, which may lead to the omission of minute details. Certain audio classification tasks that depend on subtle audio features may have trouble performing as a result of this compression.

Pros of Raw Audio Waveform-Based Approach:

- **Capturing Fine-Grained Details**

  When working with raw audio waveforms, all of the audio data, including amplitude, phase, and other time-domain characteristics, can be preserved. This may be helpful for tasks requiring the capture of minute details or the temporal dynamics of audio signals.

- **Model Interpretability**

  Raw audio waveform-based approaches frequently use more straightforward models, like recurrent neural networks or conventional machine learning algorithms. Since the feature representations in these models are more transparent and can be directly connected to audio characteristics, they can provide better interpretability.

- **Potential for End-to-End Learning**

  Waveform-based approaches enable end-to-end learning by directly processing raw audio waveforms, which enables the model to learn to directly extract pertinent features from the audio signals without relying on manual feature engineering.

Cons of Raw Audio Waveform-Based Approach

- **High Dimensionality**

  Because raw audio waveforms are high-dimensional signals, they can be difficult to process and require a lot of memory. In comparison to spectrogram-based methods, processing copious amounts of raw audio data may require more computational resources and longer training times.

- **Complex Feature Extraction**

To effectively represent audio signals when working with raw audio waveforms, it is frequently necessary to use domain-specific feature extraction techniques, such as MFCC or filter bank energies. The task of extracting pertinent features can be challenging and may call for careful planning and experimentation.

- **Limited Temporal Context**

  It may be difficult to capture long-term temporal dependencies in audio signals using raw audio waveform-based approaches. While some recurrent neural networks, such as LSTMs, can help with this, it can still be difficult to capture very long-term dependencies.

6.7 Rational behind selecting image-based classification

The decision between spectrogram-based and raw audio waveform-based methods depends on the precise specifications of the audio classification task, the available computational resources, the need for interpretability, and the characteristics of the audio data. Both strategies have advantages and disadvantages as discussed in the last section. We will go into more detail about the advantages of that approach since we heavily relied on it in this research work.

- **Interpretability**

  Spectrograms give audio signals a visual representation that makes them easy to understand. Spectrograms' color or intensity patterns can reveal the frequency content, time-varying traits, and significant elements of the audio data. This interpretability can be useful for comprehending the selection procedure and offering perceptions into the rationale behind the selection of a specific ASR model.

- **Established Image Classification Techniques**

Due to their 2D image representation, spectrograms can benefit from tried-and-true image classification methods and models. These methods, which include deep learning models like CNNs, have been thoroughly investigated and shown to be efficient for image classification tasks. One can take advantage of the vast expertise and improvements in image-based algorithms by using spectrograms, which might enhance classification accuracy.

- **Transfer Learning Capabilities**

  Spectrograms have the potential to use transfer learning methods. Spectrogram images can be used to fine-tune pre-trained CNN models, which are frequently trained on massive image datasets like ImageNet. By using this method, the ASR pipeline can take advantage of the features and patterns that have been discovered from a variety of image data, potentially enhancing the classification task's performance.

- **Computational Efficiency**

  Spectrograms can be computationally effective, particularly when compared to processing unprocessed audio waveforms. Parallelization and effective hardware resource utilization during training and inference are made possible by converting audio to spectrograms. This efficiency is especially important when processing needs to happen in real-time or close to real-time or when there aren't enough computational resources available.

- **Flexibility for Future Extensions**

  Spectrograms offer a flexible representation that can accommodate pipeline modifications or future extensions. Working with spectrograms can make it easier to integrate new image-based or computer vision techniques into the ASR pipeline, such as semantic segmentation or object detection, with the least number of changes to the current infrastructure.

6.8 Data augmentation

Data augmentation is pivotal for enhancing machine learning model performance by improving training data diversity and quality. This thesis comprehensively explores these techniques, particularly their application to spectrogram data. Augmentation methods are categorized into invasive and non-invasive approaches, spanning from classic image techniques (e.g., cropping, rotation, color correction) to advanced spectrogram-specific methods using the Librosa library. Effective techniques like pitch shifting and spectral warping are discussed, while deliberate exclusions include time stretching and additive noise due to their potential to disrupt audio structure and intricacies. This research addresses limited training data challenges, facilitating better generalization and robustness in audio-based machine learning models.

- **Categorization of Augmentation Techniques: Invasive vs. Non-Invasive**

  Augmentation techniques can be broadly categorized into invasive and non-invasive techniques. Invasive techniques involve altering the fundamental characteristics of the data, potentially introducing substantial changes. Non-invasive techniques, on the other hand, introduce variations without fundamentally altering the inherent characteristics of the data.

- **Invasive Data Augmentation Techniques**

  Invasive techniques involve modifications that substantially alter the original data. Classic image augmentation techniques, such as cropping, rotation, and color correction, fall under this category. These techniques are not directly applicable to spectrogram data due to the nature of audio data. Spectrogram data does not possess the same spatial structure as

images, rendering cropping and rotation irrelevant. Color correction, although applicable in some cases, lacks meaningful interpretability in the context of audio.

- **Non-Invasive Data Augmentation Techniques**

  Non-invasive techniques introduce variations to the data without fundamentally altering its characteristics. These techniques are particularly suitable for spectrogram data augmentation, preserving the essence of the audio. Techniques like pitch shifting and spectral warping are pivotal in this context. Pitch shifting involves altering the pitch of the audio, effectively changing its perceived tone without distorting the temporal structure. Spectral warping, by distorting the spectrogram representation, allows for controlled variations while maintaining the underlying audio content.

- **Classic Image Augmentation Techniques and Their Inapplicability to Spectrogram Data**

  Classic image augmentation techniques, such as cropping, rotation, and flip, are rooted in spatial variations. Cropping, for instance, is based on spatial coordinates, which are not meaningful in spectrogram data. Rotation and flip are irrelevant since the audio data lacks spatial orientation. These techniques, though effective for images, do not translate seamlessly to spectrogram data.

- **Advanced Spectrogram Data Augmentation Techniques: Pitch Shifting and Spectral Warping**

  Pitch shifting involves altering the frequency of the audio while maintaining its temporal structure. This technique is highly valuable for creating variations in audio data without

undermining its overall characteristics. Spectral warping, on the other hand, allows for controlled deformation of the spectrogram, introducing subtle changes in the frequency domain while preserving the temporal aspect.

- **Exclusion of Time Stretching and Additive Noise**

  Time stretching, a technique that alters the duration of audio segments, can significantly disrupt the intended structure and tempo of audio samples. This disruption can adversely affect the interpretability and meaningfulness of the audio data, thus leading to its intentional omission. Similarly, the avoidance of additive noise is driven by the aim of preserving the intricate details and nuances within the audio, safeguarding the purity of the original data.

  Data augmentation techniques are pivotal in enhancing the robustness and generalization capabilities of machine learning models. When applied to spectrogram data, techniques like pitch shifting and spectral warping prove to be highly effective in introducing variations while preserving the essential characteristics of the audio. Careful consideration and exclusion of invasive techniques, such as time stretching and additive noise, demonstrate a commitment to maintaining the integrity of the underlying audio data. As machine learning applications in audio analysis continue to evolve, the judicious selection of augmentation techniques remains paramount in driving model performance. The meticulous application of above-mentioned techniques contributes to the enhancement of audio-based machine learning models and their potential to accurately decipher the nuances within audio data.

6.9 Data splitting

Breaking down speeches into smaller segments helps in managing data effectively, leading to the generation of improved spectrograms that significantly impact model performance. Consequently, after obtaining clear stammering speech data, we delved into various parameters for segmenting the data into smaller chunks, with a specific emphasis on addressing challenges present in People Who Stutter (PWS) speech. The detailed discussion on this exploration is presented below, and for the segmentation process, we employed Audacity (Audacity, 2017) to semi-automate the division of audio files into smaller units using different parameters.

- **Silence wise data split**

  In typical speech, there is usually a brief pause between sentences, recorded as silence in speech data. This silence serves as a general guideline for approximately identifying sentence boundaries. However, individuals with stammering (PWSs) exhibit varying levels of silence and prolonged breathing in their speech profiles, indicative of the severity of their condition. Dwivedi et al. (2021) highlight silence and heavy breathing as critical features of stammering. Consequently, the conventional approach of splitting sentences based on silence in regular speech should not be applied to stammering data, as it could lead to potential data loss.

- **Sentence-wise data split**

  This parameter involves segmenting speech data according to natural language sentence boundaries. However, adopting this approach introduced a challenge related to variations in the duration of audio files. Consequently, the spectrograms generated from these speech samples exhibited diverse sizes, leading to a significant decrease in accuracy. Despite attempting to resize all spectrograms to a fixed frame, this proved impractical for both long

and short sentences, resulting in images that were either overly compressed or excessively stretched.

Another obstacle encountered during the sentence-wise data split was the manual effort required in the process. There is no readily available technique for breaking speech data based on natural language sentence boundaries. On average, a skilled individual takes about an hour to split approximately a hundred sentences. This method, however, proved neither cost-effective nor time efficient.

- **Duration-wise data split**

  We ultimately opted for the duration-based segmentation approach, considering its cost-effectiveness and time efficiency. Through our experimentation with the data, we determined that a 10-second duration effectively captures various features of People Who Stutter (PWS) speech. Consequently, we divided all speech data into audio files lasting 10 seconds each and excluded clips that fell short of the desired duration.

6.10 Spectrograms Generation

In the process of generating spectrograms, we utilized the Librosa library (McFee et al., 2015) for Python within our pipeline. All audio samples were input into the system, and spectrograms were generated from these samples (Mcfee et al., 2015). The pipeline was designed to read all audio files from a predetermined folder and then export the resulting spectrograms to another predefined folder.

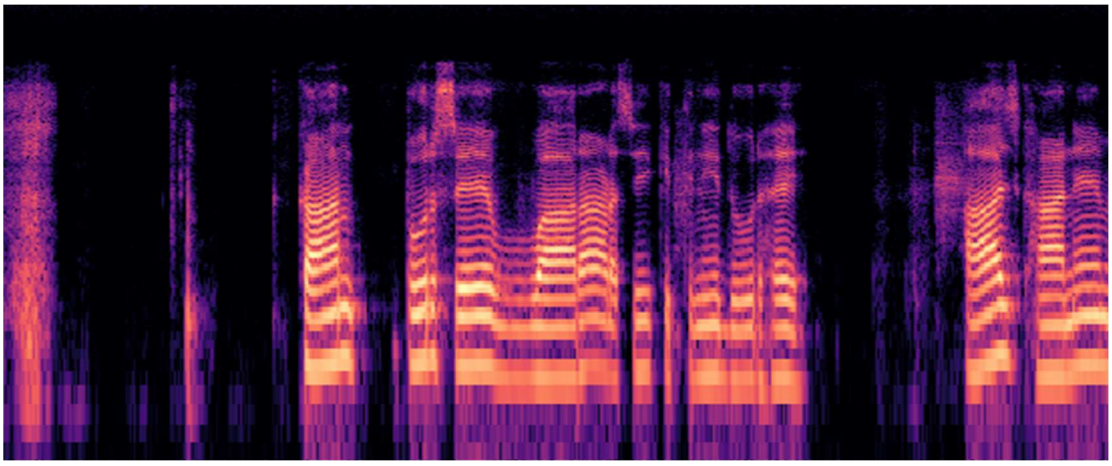

Figure 6: A sample of Spectrogram generated from the audio data

6.10.1 Training Data

To train our image classifier model for the classification task at hand, we used one hour of both stammering and non-stammering raw audio data, later converted into spectrograms. To address the limitation of our spectrogram dataset, we strategically employed data augmentation, a transformative technique. Our augmentation approach extended beyond conventional methods, tailoring techniques to the nuanced temporal and spectral aspects inherent in audio signals. By incorporating pitch shifting and spectral warping, we artfully replicated vocal nuances and frequency-domain dynamics, preserving the essence of the original audio while infusing diversity. Our comprehensive augmentation strategy deliberately excluded traditional techniques such as cropping, flipping, and rotating, as these could potentially compromise intricate audio details. Additionally, we refrained from using Time Stretching, an augmentation method altering audio duration, to avoid disrupting the intended structure and tempo of our audio samples. The omission of additive noise further exemplified our commitment to safeguarding nuanced intricacies within our audio. Emphasizing non-invasiveness, our augmentation strategy aimed to uphold the integrity

and originality of our audio data. Our training dataset comprised 6 hours, with each class featuring 1 hour of original data and an additional 4 hours of augmented data. This distribution resulted in a total of 2160 spectrograms, with each 10-second interval contributing one spectrogram. Across the 6-hour span, this translated to 360 spectrograms per hour, yielding a cumulative count of 2160 spectrograms, providing 1080 spectrograms for each class.

We implemented a data split of 80-10-10, where 80% of the data was allocated for model training, 10% for validation after each epoch, and the remaining 10% for evaluating the model's performance once it was trained. The data split was carried out following randomization, ensuring that none of the testing or validation data was included in the training dataset (Kang & Jameson, 2018).

6.11 Model architecture

The model architecture was formulated using TensorFlow's Keras API, specifically employing the Keras library (Chollet, 2015). For our model training, we utilized a Sequential Convolutional Neural Network (CNN) on both grayscale and RGB spectrogram data. The basic building blocks of our model, encompassing base layers, their corresponding activation classes, core layers, and reshaping layers along with their respective shape sizes, are outlined in the following table(Saxena, 2022). Table 7 provides a detailed representation of our model structure for the purpose of reproducing consistent results.

Layer (type)	Output Shape
sequential (Sequential)	(32, 256, 256, 3)
conv2d (Conv2D)	(32, 254, 254, 3)
max_pooling2d(MaxPooling2D)	(32, 127, 127, 32)
conv2d_1 (Conv2D)	(32, 125, 125, 64)
max_pooling2d_1(MaxPooling2D)	(32, 62, 62, 64)

Layer (type)	Output Shape
conv2d_2 (Conv2D)	(32, 60, 60, 64)
max_pooling2d_2(MaxPooling2D)	(32, 30, 30, 64)
conv2d_3 (Conv2D)	(32, 28, 28, 64)
max_pooling2d_3(MaxPooling2D)	(32, 14, 14, 64)
conv2d_4 (Conv2D)	(32, 12, 12, 64)
max_pooling2d_4(MaxPooling2D)	(32, 6, 6, 64)
conv2d_5 (Conv2D)	(32, 4, 4, 64)
max_pooling2d_5(MaxPooling2D)	(32, 2, 2, 64)
flatten (Flatten)	(32, 256)
dense (Dense)	(32, 64)
dense_1 (Dense)	(32, 3)

Table 7: Model's architecture, with the layers and their corresponding output shapes

We initiated the model with a sequential layer, where each layer processes precisely one input and produces one output tensor. Utilizing six conv2d layers, we created convolution kernels that convolve with the input layer, generating output tensors. Subsequently, each conv2d layer is succeeded by a max pooling2d layer, effectively reducing the dimensions of the feature map. The incorporation of max pooling layers serves the dual purpose of preventing overfitting and expediting the training time of our model. Following this, a flattened layer is employed to convert the data into a one-dimensional array, facilitating its passage to the subsequent layer. Ultimately, two dense layers are utilized for classifying images based on the output from the convolutional layers (Lin et al., 2017).

Activation functions introduce non-linearity to neural networks. In our model, we applied ReLU (Rectified Linear Unit) activation to each conv2d layer. ReLU is chosen for its ability to control the exponential growth of computation, mitigate the vanishing gradient problem, and enhance overall performance. For the final dense layer, we employed SoftMax activation, enabling the interpretation of the results as a probability distribution.

It is conventional to use ReLU in hidden layers and SoftMax in the output layer, as indicated in traditional practices (Asadi & Jiang, 2020).

6.12 Optimizer and loss function

In machine learning, the selection of appropriate optimizers and loss functions is pivotal for the successful training of deep neural networks. These critical components significantly influence the training process, affecting convergence speed and overall model performance(Yan, 2022).

Optimizers

Optimizers are the driving force behind the training of machine learning models, determining how neural network weights are updated during the training iterations (Rajendra et al., 2021). In my research, I have chosen to employ the Adam optimizer, a widely acclaimed choice in the machine learning community.

- **Adam Optimizer**

  The Adam optimizer, short for "Adaptive Moment Estimation," represents a significant advancement over the conventional stochastic gradient descent (SGD) algorithm. It addresses several limitations associated with SGD, making it a robust choice for training deep neural networks.

- **Adaptive Learning Rates**

  One of the key advantages of the Adam optimizer is its ability to adapt the learning rate on a per-parameter basis. This stands in contrast to traditional gradient descent methods, where a fixed learning rate is employed throughout training. Adam calculates individual learning rates for each parameter, enabling it to handle varying degrees of parameter updates. This adaptability is particularly advantageous when dealing with complex and

high-dimensional models, as it ensures that no single parameter dominates the learning process.

- **Momentum and Adaptive Scaling**

  Adam combines the principles of momentum and adaptive scaling to expedite convergence. It maintains two moving averages for each parameter: the first moment (mean) and the second moment (uncentered variance) of the gradients. These moving averages serve to smooth out parameter updates, preventing oscillations during training. Additionally, the adaptive scaling factor takes into account the scale of the gradients, further enhancing the stability and efficiency of the optimization process.

Loss Functions

Loss functions play a crucial role in the training process by quantifying the dissimilarity between predicted values and ground truth labels (Seif, 2022; Wang et al., 2022). In my binary image classification research, I have utilized two distinct loss functions: BinaryCrossEntropy and SparseCategoricalCrossentropy.

- **BinaryCrossEntropy**

  BinaryCrossEntropy is a widely employed loss function for binary classification tasks. Its primary function is to measure the discrepancy between predicted probabilities and the true binary labels, typically represented as 0s and 1s. This loss function is well-suited to my binary image classification work as it accurately quantifies the error in predicting whether an image belongs to one of two classes.

- **SparseCategoricalCrossentropy**

SparseCategoricalCrossentropy, on the other hand, is typically reserved for multi-class classification problems. It computes the cross-entropy loss between the predicted probabilities and integer-encoded class labels. While my research is focused on binary classification, it is noteworthy that if I were dealing with a multi-class problem involving more than two classes, this loss function would be a logical choice.

Comparative Analysis- Rationale for Selection

- **Adam vs. SGD**

  The decision to employ the Adam optimizer over traditional SGD is substantiated by the superior performance demonstrated by Adam across various scenarios. Adam's adaptive learning rates and incorporation of momentum and scaling factors make it an ideal choice for training deep neural networks. It alleviates the challenges associated with selecting an appropriate fixed learning rate, which can be a cumbersome task when using SGD.

- **BinaryCrossEntropy vs. SparseCategoricalCrossentropy**

  The choice between BinaryCrossEntropy and SparseCategoricalCrossentropy hinges on the nature of the classification problem at hand. Since my research pertains to binary image classification, BinaryCrossEntropy aligns seamlessly with the binary labels. However, it is important to note that if I were working on a research problem involving multiple classes, SparseCategoricalCrossentropy would be the preferred loss function due to its compatibility with integer-encoded class labels.

  Adam's adaptability in learning rates and its incorporation of momentum and scaling factors make it a robust optimizer, while BinaryCrossEntropy accurately quantifies the error for binary classification tasks. These choices reflect a thoughtful consideration of the

specific requirements of my research, ultimately leading to more efficient training and improved model performance.

6.13 Rescaling

To normalize the spectrograms, we performed a rescaling operation by dividing all pixel values by 255, transforming the pixel range from [0,255] to [0,1]. This normalization is crucial, as it prevents higher pixel value images from disproportionately influencing the model and introducing bias. Additionally, rescaling aids in promoting the convergence of our neural network model, ensuring that coefficients are adjusted within the range of [0,1], as opposed to the original range of [0,255].

- **Use of full-sized images**

  In this experiment, we refrained from downsampling our training data and instead utilized full-sized images with dimensions of 720x360. However, after 8 epochs, the validation loss exhibited an increase, indicating the onset of overfitting. The key lesson learned from this experiment was the potential drawback of using full-sized images, leading to extended training times and eventual overfitting of the model. Subsequent experiments addressed this issue by employing resized images with dimensions of 256x256. Figure 7 illustrates the training and validation accuracy, along with the training and validation loss graph for the model trained with full-sized images.

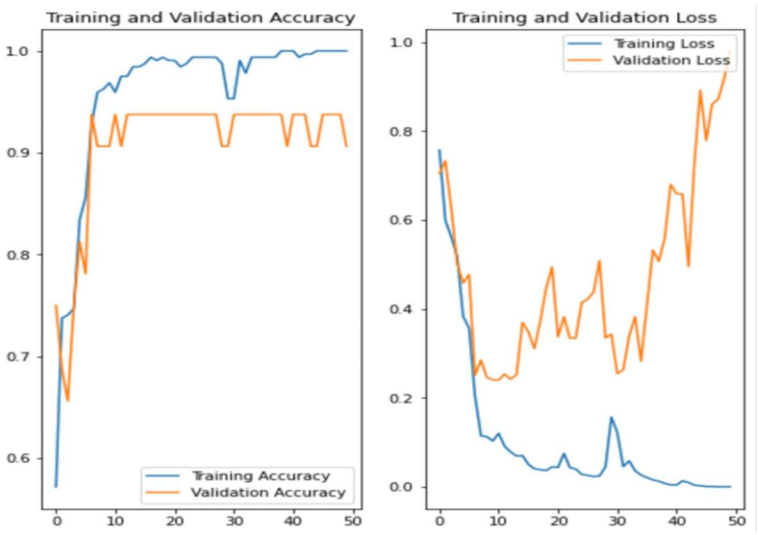

Figure 7: Comparison of training-validation accuracy and Training and validation loss.

6.14 Color as a hyperparameter

In this experiment, spectrograms were created in both RGB and grayscale formats. While maintaining consistent parameters, two distinct models were trained using RGB and grayscale datasets. It was noted that the grayscale model exhibited shorter training times compared to the RGB model. Additionally, a slight deterioration in the confusion matrix was observed. The results depicted in Figure 8 indicate that predictions made by the RGB model were more accurate, while the confidence threshold remained consistent across both models. In the results excerpt, both precision and recall scores are equivalent, attributable to the use of micro averaging instead of macro or weighted averaging. The balance in our dataset also contributed to the uniformity in precision and recall scores.

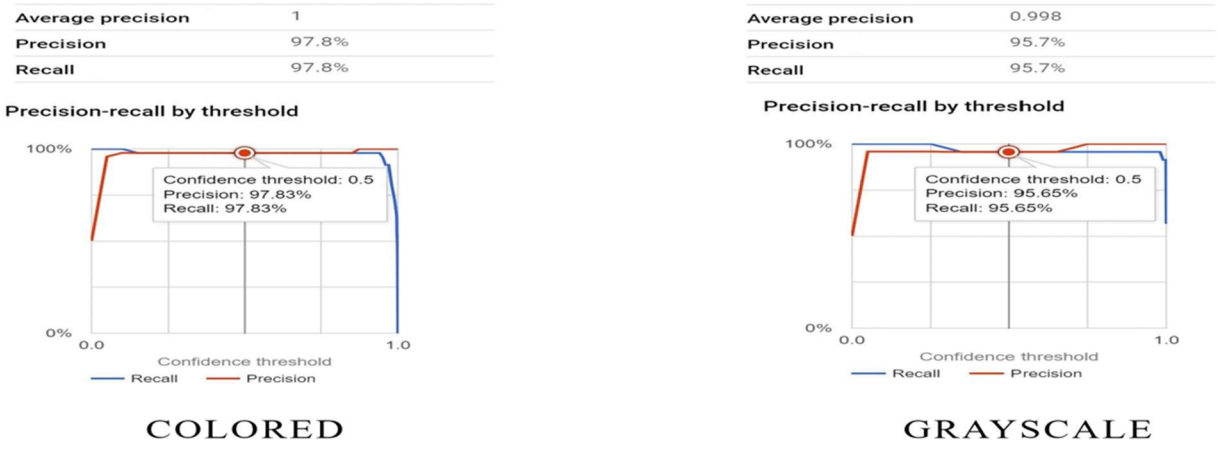

Figure 8: Comparison of average precision, precision and recall of both models

Figure 9 illustrates that the model utilizing RGB samples achieves marginally superior accuracy but demands a longer training time. Conversely, the model utilizing grayscale samples exhibits slightly lower accuracy; however, it also demonstrates reduced training time when contrasted with the RGB model. The choice between training time and accuracy can be strategically adjusted based on the specific use case.

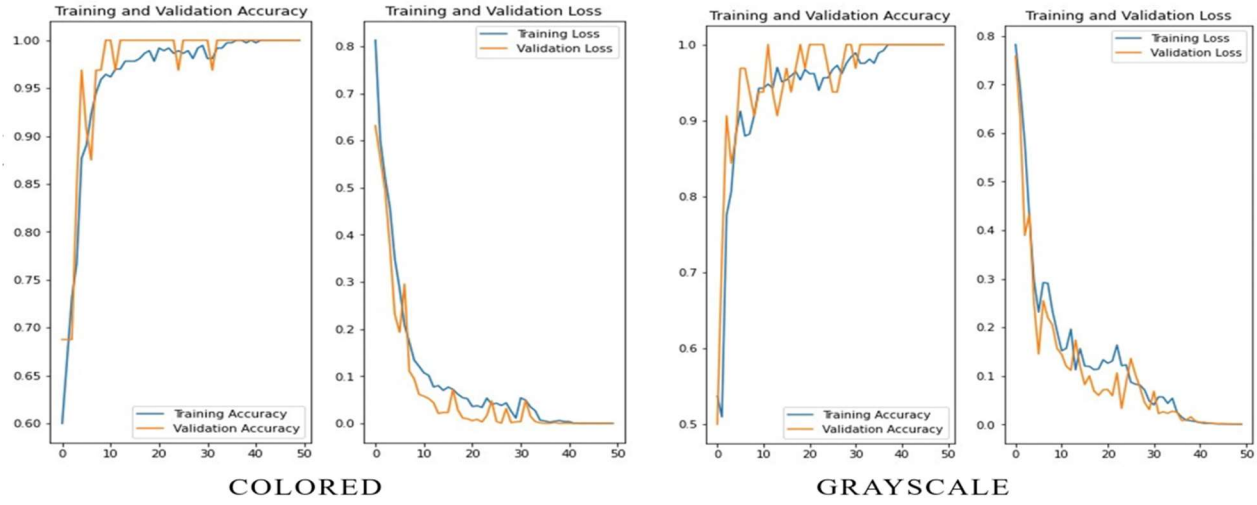

Figure 9: Comparison of training-validation accuracy, and training-validation loss of both models.

6.15 Training Patience

It is a form of regularization that enables the model to halt training if a specified number of epochs elapses without improvement in the defined metrics (Tian & Zhang, 2022). When examining the training duration of both models, we discovered that training grayscale models is faster compared to RGB models. As depicted in Figure 10, the RGB model ceased training after 42 epochs, whereas the grayscale model halted training after only 17 epochs.

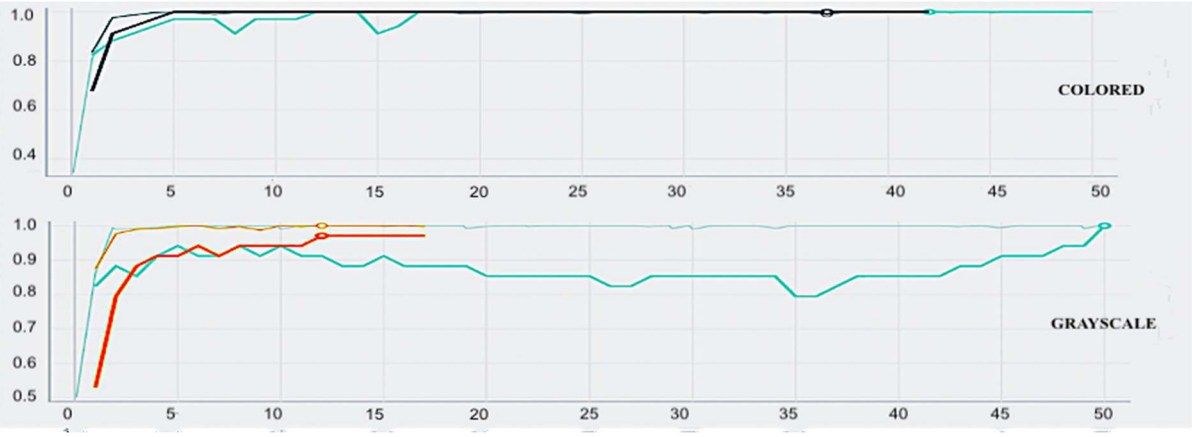

Figure 10: Comparison of training epochs of colored and grayscale model

We additionally explored variations in both training data size and training patience to assess their correlation with model accuracy. Figure 11 distinctly illustrates that training with a smaller dataset necessitated more epochs to attain higher accuracy. Conversely, as the training data size increased, we observed a reduction in the number of epochs required, attributed to the model having more parameters to learn rapidly.

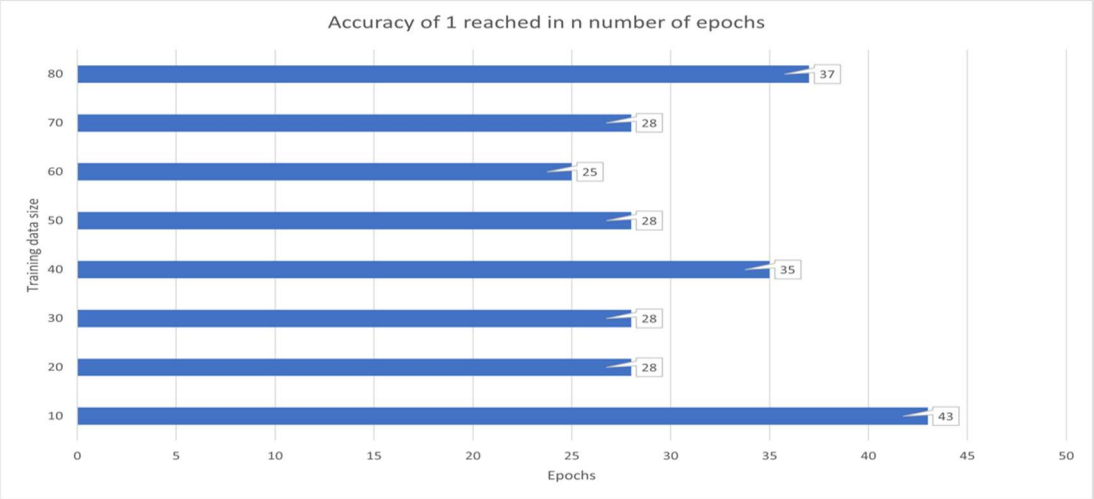

Figure 11: Correlation between training data size and number of epochs.

6.16 Gradual increase of training data

In a systematic approach, we incrementally augmented our training data while concurrently diminishing the testing data to examine the model's capability to learn from a reduced training dataset. Figure 12 reveals that, even with only 10% of the training data and 80% of the testing data, our model achieved an accuracy exceeding 91%. Throughout all eight experiments in this category, the validation data remained constant at 10%.

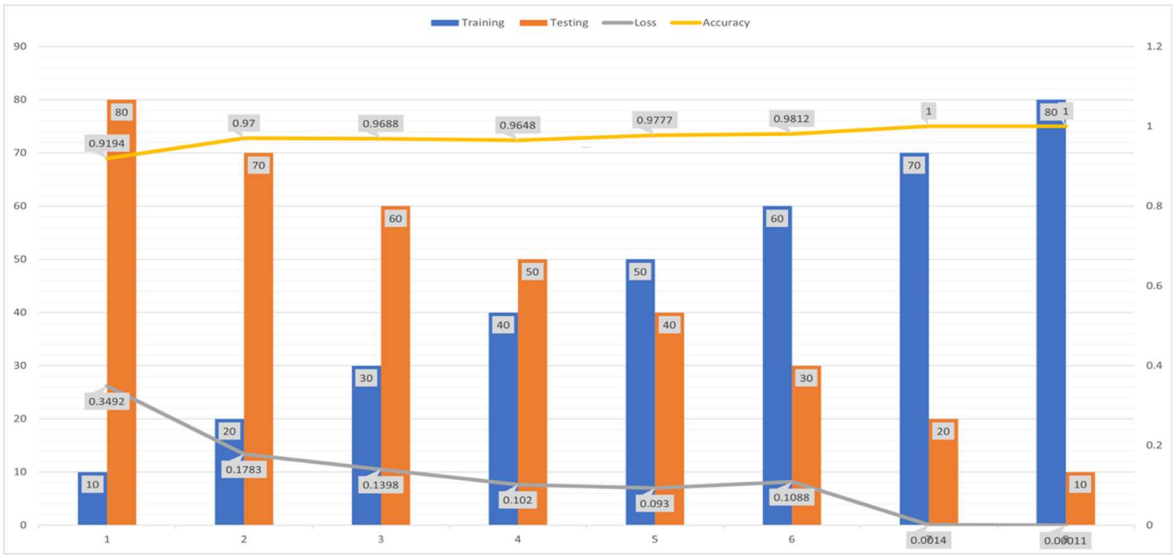

Figure 12: Accuracy results on changing the training data size

6.17 Usages

As depicted in Figure 13, we advocate for the incorporation of this binary classification model as an additional checkpoint within existing voice assistant/ASR pipelines. This strategic integration enables the processing of speech signals based on the distinctive speech characteristics of the speaker. In cases where the signal exhibits stammering traits, the speech samples can be directed to pipelines specifically designed to decipher instances of stammering. Conversely, if the speech signals lack any stammering instances, they can seamlessly proceed through conventional pipelines. Beyond the immediate identification of stammering instances, this approach opens avenues for tailored interventions, providing valuable support to individuals with speech impediments. Furthermore, the integration of our model enhances ASR technologies, fostering a comprehensive approach to speech analysis that accommodates a diverse range of speech characteristics. Given the pivotal role of speech recognition in various domains, our proposed model architecture has the potential to significantly enhance user experiences and broaden accessibility.

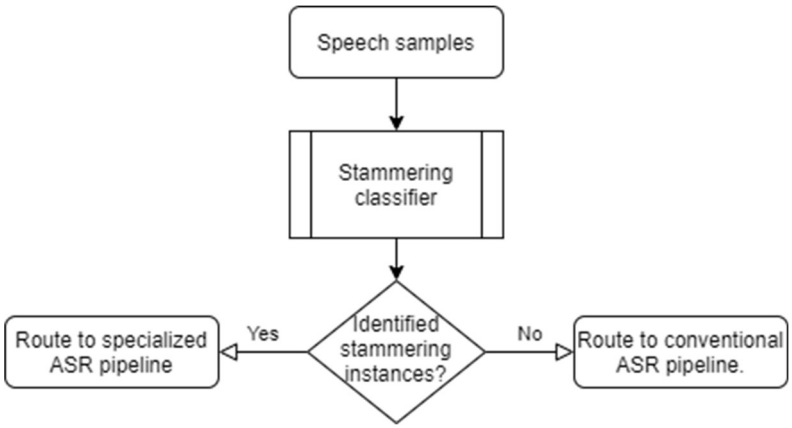

Figure 13: Binary classifier pipeline

6.18 Results

Our efforts to create a binary classifier aimed at detecting stammering patterns in Hindi speech have produced positive outcomes. There is a clear demand for increased research in Natural Language Processing (NLP), particularly in the realm of Hindi Speech-Language Pathology (SLP), considering the current challenges posed by limited resources and data availability for applied research. The notable accuracy observed in various machine learning experiments utilizing Hindi stammering data highlights the potential of employing a binary stammering classifier to enhance the performance of existing Hindi Automatic Speech Recognition (ASR) systems. As depicted in Figure 14's confusion matrix, our classification experiments achieve accuracy exceeding 95% with a test dataset comprising over two hundred samples.

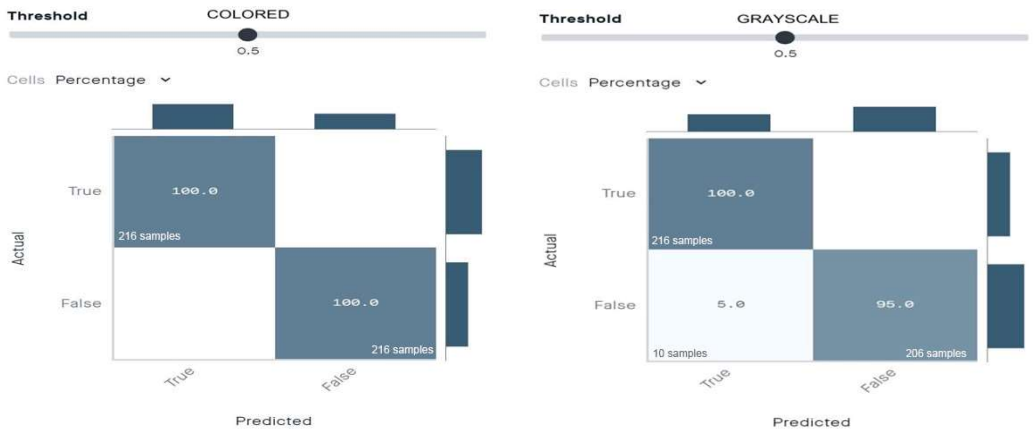

Figure 14: Confusion matrix derived from our experiments involving the top-performing colored and grayscale models

6.19 Wake word

In the domain of voice-activated technologies, the wake word, often interchangeably referred to as a trigger word, wake-up word, or hotword, stands as a fundamental and technically intricate element. This unassuming linguistic cue plays a critical role as the point of initiation for voice-driven interactions with artificial intelligence-based virtual assistants like Alexa, Siri, and Cortana. Beyond its apparent simplicity, the wake word conceals intricate layers of complexity. We try to cover the significance of wake words, elucidating their indispensability within the dynamic landscape of voice technology. We also delve into the technical intricacies that underlie the functionality of wake words, shedding light on their role as the point of ingress for users, facilitating seamless and intuitive engagements with the digital sphere.

- **Significance of Wake Words**

  In the context of voice-activated technologies, the significance of wake words cannot be overstated. These linguistic cues serve as the linchpin of user engagement, effectively acting as the initial trigger for voice interactions with digital assistants. They encapsulate the bridge between human intent and machine responsiveness, offering users an intuitive

means to initiate communication. Beyond their role in facilitating user-device interactions, wake words are paramount in addressing privacy and security concerns. By ensuring that the virtual assistant remains in a passive listening state until explicitly activated, wake words mitigate the risk of accidental eavesdropping, thus enhancing user confidence. Additionally, they contribute to operational efficiency by conserving resources, including power and processing capacity, as the device need not be in an active listening mode continuously. Furthermore, the personalization aspect of wake words, which often allows users to customize them, fosters a sense of ownership and affinity, thereby enriching the user experience.

- **Technical Intricacies of Wake Words**

  The technical intricacies underpinning the functionality of wake words are rooted in advanced signal processing, acoustic modeling, and machine learning techniques. At the core, wake words necessitate continuous audio sensing, wherein the device continuously monitors ambient audio for acoustic patterns that closely resemble the predefined wake word. This monitoring is facilitated by sophisticated neural networks and acoustic models trained to recognize phonetic characteristics. Upon detecting a potential wake word candidate, keyword spotting algorithms come into play, verifying the match with the predefined word while filtering out false positives. Notably, contemporary voice-activated virtual assistants, such as Alexa, Siri, and Cortana, augment wake word detection with contextual understanding. They analyze the surrounding dialogue to grasp user intent and provide relevant responses. This multifaceted process, deeply rooted in signal processing and machine learning, enables the precise and accurate activation of virtual assistants upon hearing the wake word.

- **Transformative Impact on the Field of Voice-Activated Technologies**

  The incorporation of wake words has ushered in a transformative era in the realm of voice-activated technologies. These linguistic cues have served as a catalyst for human-computer interaction, rendering it more natural, accessible, and efficient. By establishing a clear and intuitive entry point for users, wake words have played a pivotal role in mainstreaming voice-activated devices and services. Their contribution to user privacy, through the prevention of inadvertent activations, addresses one of the most critical concerns in the adoption of voice technology. Additionally, the resource efficiency achieved through wake words has led to more sustainable and practical implementations of voice-activated devices. The personalization aspect has not only enhanced user satisfaction but also encouraged greater user engagement. Ultimately, wake words, with their technical sophistication and user-centric design, have redefined the landscape of voice-activated technologies, paving the way for continued innovation and integration in diverse facets of daily life.

6.19.1 Problems with Wake Words

While wake words have undeniably revolutionized voice-activated technologies, they are not without their challenges and limitations. These issues encompass both technical and user experience aspects:

- **False Positives and Negatives**

  One of the persistent technical challenges associated with wake words is the occurrence of false positives and false negatives. False positives involve the erroneous activation of the virtual assistant when a non-intentional phrase closely resembles the wake word.

Conversely, false negatives occur when the wake word is not recognized, leading to missed user commands. Achieving a balance between sensitivity and specificity in wake word detection remains a complex task.

- **Privacy Concerns**

  Despite their role in enhancing privacy by preventing constant eavesdropping, wake words raise privacy concerns of their own. The requirement for the device to be in a constant listening state to detect the wake word raises questions about data security and potential misuse of audio recordings. Users are increasingly wary of the data collected through wake word activations.

- **Dialect and Accent Variations**

  Wake word recognition systems may struggle with dialectal and accent variations, leading to reduced accuracy for users with non-standard speech patterns. Ensuring inclusivity and accuracy across diverse linguistic backgrounds remains an ongoing challenge.

- **Sensitivity to Noise**

  Ambient noise can interfere with wake word detection, leading to false activations or missed wake words. Achieving robust performance in noisy environments is a significant technical hurdle.

- **Limited Personalization**

  While some virtual assistants allow for wake word customization, not all platforms provide this feature. Limited personalization options may lead to less user satisfaction, particularly for individuals who prefer a more personalized interaction with their devices.

6.19.2 Gaps in Wake Word Technology

The development and implementation of wake words in voice-activated technologies also reveal certain gaps and areas for improvement:

- **Multimodal Wake Words**

  Integrating multiple input modalities, such as gestures or visual cues, alongside wake words could enhance user experience and accessibility, particularly in situations where voice commands may be impractical or inconvenient.

- **Robustness in Adverse Conditions**

  Further research is needed to improve the robustness of wake word detection in adverse conditions, such as elevated levels of background noise or unusual acoustic environments, to ensure consistent performance.

- **Language Support**

  Expanding language support beyond the commonly spoken languages remains a challenge. Ensuring that wake words are accessible and effective in a broader linguistic context is essential for global adoption.

- **Privacy Enhancements**

  Addressing privacy concerns by developing more transparent and secure wake word architectures, such as on-device processing, can enhance user trust in voice-activated technologies.

- **User Experience Optimization**

  Continual efforts are needed to optimize the user experience surrounding wake words, including minimizing false activations, and providing more robust feedback to users when wake words are recognized or not.

While wake words have brought about remarkable advancements in voice-activated technologies, they confront challenges related to accuracy, privacy, and user experience. Addressing these problems and bridging the existing gaps in the wake word technology remains pivotal to the continued evolution and acceptance of voice-driven interactions.

6.19.3 Proposing a "Sleep Word" for Improved Voice Assistant Functionality

The inception of the "Sleep Word" concept was rooted in a meticulous data collection process focused on individuals who stammered or grappled with speech impediments. Throughout this investigative phase, a pronounced and recurrent issue came to the fore – voice assistants would often prematurely disengage during instances of speech blockages or pauses. This observation profoundly underscored the need for a novel solution to mitigate these interruptions and enhance the user experience for individuals with speech disorders.

The "Sleep Word" concept was born from the pressing challenge identified during data collection. It essentially acts as a deliberate pause mechanism, initiated by users when needed. This mechanism grants users greater control over voice assistant interactions, effectively addressing the challenges posed by speech disorders. By allowing users to temporarily suspend voice assistant responses until they are ready to continue, “Sleep Word" provides an invaluable tool for individuals who experience speech impediments.

Crucially, the idea for the "Sleep Word" was not conceived in isolation but was a direct response to the practical experiences and needs of users with speech disorders. The goal

was twofold: to eliminate the frustration caused by voice assistant interruptions and to create a more inclusive and accessible voice technology environment. In essence, the "Sleep Word" concept evolved from empirical observations, aiming to empower users and enhance the usability of voice-activated technologies, irrespective of speech-related challenges they may face.

In the landscape of voice-activated technologies, the concept of a "Sleep Word" emerges as a potential solution to address aforementioned issues in real-world usage scenarios. This novel idea introduces a trigger mechanism that empowers users with greater control over when their voice assistant listens and responds, particularly catering to those with speech disorders, such as stammering or other conditions that result in pauses and blockages during speech. The proposed "Sleep Word" operates as a complementary counterpart to the well-established wake word, with the ability to temporarily halt voice assistant listening until explicitly deactivated. This innovation can significantly enhance user experience and accessibility in voice-activated technology contexts.

6.19.4 Key Features and Functionality

- **Sleep Word Activation and Deactivation**

  Users would have the capability to activate the "Sleep Word" at their discretion. When the "Sleep Word" is uttered, the voice assistant would temporarily cease listening and processing commands until the "Sleep Word" is deactivated. This offers users full control over when their device responds to voice commands.

- **Addressing Speech Disorders**

  A primary and crucial use case of the "Sleep Word" is its utility in accommodating individuals with speech disorders, particularly those who stutter. In cases where a user experiences a blockage or pause due to their condition, the "Sleep Word" serves as a practical means to prevent premature command processing by the voice assistant. This ensures that users with speech disorders can complete their intended instructions without interruptions, eliminating potential frustrations and inaccuracies stemming from the misinterpretation of partial commands.

- **Customization**

  Recognizing that different speech disorders and individual needs vary significantly, the proposed "Sleep Word" system offers users the option to define their own unique "Sleep Word" based on their specific speech patterns and requirements. This customization empowers users to tailor the functionality to their unique circumstances, ensuring a more accurate and personalized user experience.

- **Predefined Sleep Words**

  To offer a degree of convenience for users who may not wish to define their own "Sleep Word" or are unsure of what to choose, a selection of predefined "Sleep Words" could be made available. Users can activate any of these predefined options to promptly implement the "Sleep Word" functionality.

- **Enhanced Accessibility**

  The introduction of the "Sleep Word" feature represents a significant step toward improving the accessibility of voice-activated technologies. By addressing the unique needs of individuals with speech disorders, it enables marginalized users to participate fully

in the benefits of voice-assisted interactions, effectively demarginalizing them within the context of voice technology.

The introduction of the "Sleep Word" concept presents a promising avenue for advancing the functionality and inclusivity of voice-activated technologies. This innovative feature not only grants users greater control over their devices but also provides a tailored solution to the challenges faced by individuals with speech disorders. Customizability, combined with the option of predefined "Sleep Words," ensures adaptability to diverse user needs, fostering a more accessible and equitable voice technology ecosystem. Further research and development in this direction holds the potential to significantly enhance user experience and broaden the scope of voice-assisted interactions.

6.20 Conclusion

In this chapter we discussed computational experiments, along with the core components that constitute the foundation of our research on stammering binary classification, particularly within the context of Hindi language data. Our exploration delved into Machine Learning, where Classification methods emerged as the bedrock of our analytical framework. Leveraging the transformative power of Spectrogram conversion, we seamlessly bridged the gap between audio signals and image classifiers, providing a unique perspective that enhanced the precision of our model. We presented a comprehensive analysis of the associated pros and cons of Spectrograms. This chapter further discussed our research with a detailed examination of data augmentation methods, unveiling the strategies employed to bolster the robustness of our model. The architecture of our binary classifier was crafted through multiple experiments featuring parametric variations, we sought to optimize performance and accuracy. As our experiments unfolded, results showed the efficacy of our approach. Yet, our contributions extend beyond classification. In a pioneering

move, we introduced the concept of a "sleep word" to run in parallel with traditional "wake words" in voice-activated systems. This innovative idea holds promise for individuals with speech conditions, opening new avenues for inclusive technology design.

In the next chapter, we will engage in deep discussions, critically assess limitations, explore potential applications, and draw meaningful conclusions from the research work. We explore the broader implications of our research, reflecting on its significance, and envisioning the pathways it paves for the future of assistive technologies and inclusive design.