The Articulation of Speech: A Deep Learning based Lip and Voice Recognition

by

Munia Shaheen
20101050
Akib Zabed Ifti
23341129
Ariful Hassan
20301259
Junaed Hossain
20101196

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
September 2023

© 2023. Brac University
All rights reserved.

Abstract

Understanding speech just through lip movement is known as lipreading. It is a crucial component of interpersonal interactions. The majority of the previous initiatives attempted to address the English lipreading issue. However, our goal is to build up a deep neural network for the Bangla language that can produce comprehensible speech from silent videos just by capturing the speaker's lip movements. Despite the fact that there is research on this topic in various languages, Bangla does not currently have a study or a suitable corpus to conduct research. Therefore, our research aims to investigate the performances of the state-of-the-art deep learning models on the Bangla corpus and build our own deep learning model suitable for Bangla lip-reading.

Keywords: Lipreading. Deep learning, Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Lip feature extraction, Lip region localization.

Nomenclature

LR: Lip Reading

BLR: Bengali Lip Reading

DL: Deep Learning

CNN: Convolutional Neural Network

RNN: Recurrent Neural Network

ASR: Automatic Speech Recognition

VSR: Visual Speech Recognition

BSR: Bengali Speech Recognition

BVLDB: Bengali Visual Lipreading Dataset

BLSD: Bengali Lipreading Speech Dataset

WER: Word Error Rate

CER: Character Error Rate

Accuracy: Accuracy of lip reading predictions

F1-score: F1-score for lipreading accuracy

Acknowledgement

Firstly, all praise to the Great Allah for whom our thesis has been completed without any major interruption.
Secondly, to our supervisor Dr. Mohammad Iqbal Hossain, and co-supervisor Mr. Rafeed Rahman for their constant support and advice in our work. He helped us whenever we needed help.
And finally, all the people who contributed to our dataset and gave us their consent to use their videos. Without their support, our research would not be possible.

Table of Contents

List of Figures

Chapter 1

Introduction

Lipreading is the technique of comprehending what someone is saying only by their lip movement. Humans have long been able to read lips manually. However, automating lipreading is a significant goal since human lipreading performance is weak and constrained. Initially, systems for reading lips were created using traditional machine-learning techniques. Deep learning applications' prominence, particularly in recent years, has caused this topic to be studied more than in the past. The main components of automated lip-reading include face recognition, lip localization, feature extraction, classifier training with corpus, and word/sentence recognition by lip movement.

Nevertheless, due to challenges in the extraction of spatiotemporal features, generalization across speakers, and environmental noise, lip-reading is an extremely difficult task. Additionally, there are a lot of homophones, which implies that the same lip movement can produce a variety of different characters. Furthermore, some of the phonemes are created inside the throat and mouth and cannot be identified by simply observing a speaker's lips. Also, lip-reading from video without audio or text data depends on a variety of factors, including lighting, recording distance, and the speaker's speaking style. The endeavor becomes more challenging by external noises, mumbling sounds, and guttural sounds. Therefore, lipreading is a challenging task for both humans and machines.

Even though lip reading is a challenging technique to process, it is essential in a variety of fields, including accessibility, noise-sensitive communication, human-computer interaction, multimodal communication, security, and numerous more fields. Lip reading and voice recognition technologies can greatly improve accessibility for the deaf. Additionally, in noisy situations where speech could be hard to hear, such as crowded public spaces or industrial settings, lip reading technology might improve communication. These technologies enable individuals to understand speech

even in challenging acoustic environments, improving the effectiveness and precision of communication in commonplace applications. Recent years have seen an increase in enthusiasm for studies on the security of merging lipreading with biometric data, such as biological fingerprints or facial recognition. Studies that use lip movement monitoring for individuals include printing messages to smartphones in loud circumstances using visual data instead of auditory data and implementing various security measures utilizing visual silent passwords[27-29]. The applications of lip-reading technology are mentioned below:

- Helping hearing-impaired people.
- Transcribing old silent movies or videos.
- Facilitating the alert for public safety.
- Improving audio in existing videos.
- Enabling video conversation in noisy or silent places, like libraries.
- Identifying utilizing biometrics.
- Synthesizing simultaneous voice from multiple speakers.
- Enhancing automatic speech recognition's overall
- Dictating information or messages into a phone while surrounded

1.1 Problem Statement

In today's world lip-reading is needed in various fields and both humans and machines are able to do some sort of lip reading. However, human lipreading performance is poor and limited. For example, studies showed that people with hearing impairments only attain an accuracy of 17±12% for a small subset of 30 monosyllabic words and 21±11% for 30 complex words. Thus, automating lipreading is a key objective. While classical machine learning techniques can be used to read lips, they are slower and less accurate than deep learning techniques. These days, with the use of deep learning, it is liable to translate lip motions into relevant words.
Even though there has been little research on lipreading in English, currently, there is no known research on lipreading in the Bangla language. Even there is no known corpus in Bangla suitable for this task. There are 265 million native and non-native speakers of the Bangla language worldwide, making it the fifth most widely used language. A significant majority of them are unable to speak English. Because of this, it will be challenging for them to use present technologies that require English.

In our literature review, we will see most of the research used on English lip-reading becomes useless when trained on different language datasets. Therefore, research in lip reading in the Bangla language is crucial for the development of this field.

1.2 Aims and Objectives

1. Creating a corpus that can be used in further research in the Bengali language.
2. Finding the feature extraction method for the Bangla corpus.
3. Examining the results of the current best models used in the English corpus using our Bangla corpus.
4. Establishing the ideal lip-reading model for the Bangla language

Chapter 2

Literature Review

In this literature review, we will be discussing the previous research in this domain of lip-reading. Most of the research used English speakers in their corpus. However, there are few types of research in German, Urdu, Korean, Chinese, and Turkish which gave state-of-the-art performance in their own language datasets. In our literature review, we will review lip-reading in English, Urdu, Chinese, Turkish, and German. As there is no known research on lip-reading in Bangla, we will review a few papers on Bangla [1] voice recognition systems on the existing technologies.

2.1 Language: English

The first hand-segmented phone-based sentence-level lip-reading research [2] was released in 1997 and used Markov models (HMMs) in a small dataset. Later, using an HMM in conjunction with hand-engineered features in the IBM ViaVoice dataset, another researcher conducted the first sentence-level audiovisual speech recognition [3] . Additionally, the authors train an LDA-transformed version in an HMM/GMM system [12] using the GRID corpus, outperforming the prior state-of-the-art with an accuracy of 86.4%. However, in these automated methods, extraction of motion information and generalization across speakers are both regarded as a big issue [10] .

In automated lip reading, deep learning is needed as such tasks necessitate intensive preprocessing for an image or video frame extraction or other forms of manually created vision pipelines[5] [8]. However, deep learning is seldom used in lipreading and only performs classification, not sentence-level sequence prediction in most of the research.

The paper [11], is the first known sentence-level lipreading model that uses the GRID corpus [7] consisting of 1000 sentences of audio and video by 34 speakers.

In this paper, the DLib face detector, iBug face landmark predictor [9], and affine transformation were used for preprocessing. Here, they introduced a new architecture called, LipNet. In the LipNet architecture, the input is processed by 3 layers of STCNN (Spatiotemporal convolutional neural networks) which is followed by a spatial max-pooling layer. Two Bi-GRUs then follow the retrieved features. Then at each time step, a linear transformation is applied before performing a softmax over the vocabulary augmented by the CTC (Connectionist Temporal Classification) blank, followed by the CTC loss. The authors evaluated the performance of the LipNet model using the word error rate (WER) and character error rate (CER). The LipNet model is the first sentence-level lip reading that gives 95.2% accuracy at the sentence level outperforming a human lipreading baseline and exhibiting better performance than the word-level state-of-the-art in the GRID corpus. However, this paper has some limitations, the model only works for sentences consisting of 6 words and to improve their research they need larger datasets.

The paper [13], used the same GRID corpus [7] used in the paper LipNet [11]. To begin with, they trained an autoencoder of audio files. A CNN-LSTM architecture is connected to a decoder of a pre-trained autoencoder, while the main network uses a 7-layer 3D convolutional structure to extract spatiotemporal information from the input video sequence. The core lip reading network was trained with batches of 32, and data augmentation was carried out by either flipping images horizontally or adding reasonable amounts of Gaussian noise to the data. For evaluation, they compared their result with Vid2Speech [14] in terms of PESQ(Perceptual Evaluation of Speech Quality) [4], Corr2D, and STMI(Spectro Temporal Modulation Index) [6] and gave better results. This paper showed ways to improve the audio quality for better prediction with a 98% correlation. However, they only used the GRID dataset, which contained only 6 sentences with a small word probability, to evaluate their model. For future work, more train data needs to be gathered, speech reconstructions need to include emotions, and an end-to-end framework needs to be built.

Next, in the paper [29], the author introduces a novel method for text identification from a silent lip movement video where they used the GRID corpus [7]. Here, A different technique for data augmentation is applied to the dataset for each epoch: by randomly inserting photos from the dataset video and adding modest quantities of Gaussian noise, we can simulate real-world conditions. The auditory MFCC features are converted in the architecture to a variable-length sequence of video frames via the visual-to-audio feature architecture. Second, the text information is distinguished from the audio feature by the architecture of the audio feature to text. A 7-layer 3D convolutional network (CNN) is used to recover the spatial and temporal

characteristics of the video stream. Their research revealed that the model suggested in this article is capable of 92.76% validation accuracy. This paper has the same limitations as the first two papers as they used GRID corpus.

This paper [20], proposes a deep-learning technique for speech enhancement. A wellestablished GRID [7] [20] and ChiME3 [20], [17] corpus have been used as datasets for this method. The authors used a speech enhancement framework to extract audio features from a noisy speech by Enhanced Visually-Derived Wiener Filter(EVWF). Then they enhanced this wiener filter by proposing LSTM-based filter bank estimation. After that, they worked on a regression model for lip reading where they used LSTM & MLP. From a video, they extracted audio and frames, and with the help of hamming window and object tracker, lip cropper they managed to get audio and video features successfully. With one frame the LSTM model MSE of 0.092. While, with 18 frames MSE of 0.058 has been achieved. Again, With one frame MLP-based lip reading model achieved MSE of 0.199 only and 0.209 MSE while working with eighteen frames. LSTM can safely find out audio features which are clean but vanilla neural network models can not accomplish this task as smoothly as LSTM. At low SNR the model can give a standard pl-score but at high SNR, it outperforms but still works better than conventional speech enhancement approaches. However, the result and accuracy could be much higher, a context-aware AV algorithm can be used also and the dataset is much smaller to train as they have selected fewer frames while training.

The research [25] conducted small research where the two steps of the traditional lipreading method are feature extraction and categorization. In this paper[25], they did not mention the dataset, however, they used Speech recognition using a Convolutional Neural Network. This model is trained to operate on large amounts of multi-dimensional data. After that, Dynamic lip videos, CNN, HMM, and decoding are used to extract features and appropriate words from HMM models. To compare artificial neural networks for voice identification, CNN and HMM were utilized for visual feature extraction and frame-level features, respectively. With an accuracy rate of 80words from a series of photographs of the lip region. RNN has certain drawbacks like explosive issues and it cannot process for a long sequence, which is the main restriction of this paper. RNN training is a challenging task as well. Research could lead to a speaker-independent recognition system in the future.

In this paper [22], the authors worked only with the micro-content of the English language which is the alphabet. They built the dataset having 20 letters and more than 2700 recorded videos which are a maximum of 2 seconds long. They prepared

the data by converting videos into frames, detecting human faces, detecting mouths from the faces, and selecting the keyframes. The model used in this paper is CNN and stochastic gradient descent (SGD) where CNN is used for recognizing letters and also extracting features. In this paper [22], VGG19 pre-trained CNN has been used containing sixteen convolution layers, five max-pooling layers, and more than two fully connected layers. Finally, after testing, models can predict those twenty alphabets 95% of the time for the testing set whereas an accuracy of 98% for the testing set. Letters that have similarities with other letters were conducted with an accuracy of less than 99% and the letters of distinguished features managed to have over 99% accuracy. The authors only worked with letters only, word or sentence prediction from videos was not their concern in this dataset. Also, the dataset is much smaller for training which consisted of just 20 alphabets but at the same time they worked with so many videos so this could be much more in number.

In contrast to earlier published approaches for lip reading that are based on sequence to-sequence architectures, a research paper [23] based on Deep Convolutional Neural Networks produced a hybrid lip-reading (HLR-Net) model from a video. In this paper GRID dataset [7] is used to conduct sentence-level lip reading. Here, HLR-Net was compared to LipNet, LCANet, and A-CTC models for unseen and overlapped speakers. The suggested model is composed of a CNN model, an attention layer, and a CTC layer. The proposed method is based on the attention deep learning model and converts a video of lip movement into frames using OpenCV before extracting the mouth component with the Dlip package. While the decoder employs attention, fully connected, activation function layers, the encoder makes use of inception, gradient, and bidirectional GRU layers. With a CER of 4.9

The authors of [26] attempted to compare all currently available deep learning architectures and additional techniques offered by other researchers in order to determine which could offer the best performance. The suggested research project uses the MIRACLVC1 [16] dataset, which contains 3000 occurrences of words and phrases, with 150 examples in each folder. 10 words and 10 phrases are spoken by five men and ten women in each piece of data and each word/phrase was spoken 10 times. The Dlib facial detector and Dlib mouth detector were used in this research to extract the image's mouth and identify the speaker's face. AlexNet, VGG16, Inception V1, ResNet50, Auto-Encoder, Q-ADBN, and EfficientNetBO are used to train the model. EfficientNetBO will serve as the model's foundation in this design, to which the Attention block and LSTM layer are added to produce a finer performance. The research had an accuracy of 80.133% using EfficentNetBO and 86.67% with EfficientNet and Attention Block. Finally, using LSTM on top of that gives an

accuracy of 91.13%. For their future work, lip movement detection can be improved to include additional words and phrases from different speakers.

In this paper [28], the authors developed their model by extracting the mouth area and segmenting the area from a frame by using a hybrid model. For this, they have used the LRW dataset [19] from BBC TV broadcasts having audiovisual speech segments. Each video has 29 frames from which 22 frames have been selected. Their suggested method of segmentation has been split into two different stages. First, detecting the mouth and then applying an improved hybrid model. The models they used for lip reading are Bi-GRU which has two hidden layers and 2D CNN. First, they wrap the entire CNN input model in the TD function, then pass it into the Bi-GRU layer for extracting comprehensive information from previous layers' characteristics. Afterward, the outputs have been infiltrated into a pooling layer, and then all the outputs are connected into a softmax function to produce every word's possibilities. Moreover, by using ReLU activation functions two convolutional layers have been added. That's how they omit the problem of vanishing gradient. After 6 epochs loss approaches 0 in training data and 0.4 in testing data. On the other hand, with segmentation lip reading gives 90.385% accuracy and without segmentation, it gives 85% accuracy. So, their proposed architecture is doing great work on classifying lip motion sequences on words. However, in this paper, they have not worked with sentences. Also, the authors have not mentioned the sequence of the words

2.2 Language: Urdu

Few studies on lip reading in other languages have been published. The paper[18] shows lip-reading for the Urdu language that uses a corpus containing ten words. As there was no available corpus for Urdu lip-reading, the authors created a corpus of 10 words repeated 10 times by 10 participants. The recorded videos are preprocessed by cropping them to a standard size, applying the Viola-Jones face detector, and a mouth detector, and then cropping each video. To train the dataset, at first, they used LipNet [11], which failed to train due to CTC loss. Then they used RNN and LSTM to train the model on Urdu words and digits and got better accuracy from the LSTM model which is 62% and 72% respectively for words and digits. To show how effectively audio-visual lipreading performed in noisy conditions, the researchers in this study individually trained two different models for audio and video. However, they were unable to combine both networks. Additionally, they presented a small corpus of Urdu words which worked as a classification task with no sequence. Finally, they hope to create a model in the future that can predict text sequence and is akin to lipreading.

2.3 Language: Chinese

The first sentence-level Chinese lipreading was introduced in the paper [21]. In this study, they created their own dataset from CCTV News and Logic Show, labeling it with Hanyu Pinyin (a phonetic interpretation of Chinese), and end up with 349 classes and 1705 characters. They present a unique two-step network for sentence-level Mandarin lipreading where predicting the probability of Hanyu Pinyin is the first step. A max-pooling layer is applied after a 3D convolution that suggests a series of frames as input. Then, more visual features are extracted using DenseNet. The visual features are processed using a two-layer resBi-LSTM, which is subsequently accompanied by linear and softmax layers. The complete network is then trained using the CTC loss function. The second phase involves using a stack of multi-head attention to translate Hanyu Pinyin into Chinese characters. The suggested network has an absolute 13.91% and 14.99% rise in Hanyu Pinyin and Chinese character accuracy compared to LipNet [11], whereas ResNet [15] technique has an increase of 4.68% and 4.74%. The problem that remains is that for different words Hanyu Pinyin might be the same, but Chinese characters would be different due to the different contexts which indicates that Hanyu Pinyin is unable to capture context.

2.4 Language: Turkish

In the paper [27], two new datasets were created for the Turkish language, one with 111 words and the other with 113 sentences. In the paper[27], the Media-pipe framework determines the lip position and removes it from the image. Using a convolutional neural network (CNN), features are first retrieved to perform lip detection and then the classification process is completed using bidirectional long short-term memory(Bi-LSTM). The results of the experiment demonstrate that, for words and sentences, respectively, ResNet-18 and Bi-LSTM pair offer the best results, with accuracy scores of 84.5% and 88.55%. However, the limitation of the paper is that each speaker is positioned at 1.5 m in these photos, and the video clips taken to generate the dataset were captured in identical lighting and ambient conditions.

2.5 Language: German

In this paper[17], the researchers worked on the German language. They took the "Lip Reading in the Wild" dataset created by Chung Zisserman(2016). First of all, they used English for comparison and transfer learning. Then, they brought up the issue of copyright and data protection in Germany. The dataset consists of MPEG4 format videos which are 1.16 seconds long. In the processing part, they cropped the videos into 96*96 pixels only focusing on the lips of the speaker using the Glips model. As the videos have audio too, the video and audio files were stored in separate files and synchronized in the last step. For the subtitle part, they used WebMaus to synchronize with the audio. They did two experiments in this paper [17]. The validation accuracies of GLips and LRW in experiment 1 are respectively 78.4% and 77.8%. The validation accuracies of the transfer-learned models were 82.2% and 81.6% respectively. The LRW network's average validation accuracies in experiment 2 were 79.2% and 80.4%, respectively. The average validation accuracies of the GLips networks were 78.4% and 79.6%, respectively. The fact that learning was successfully transferred between the two languages suggests that there are aspects of lip reading in both datasets that are linguistically independent and can be applied to different languages. We assume that the difference between the models trained on GLips and LRW lies in the quality of the data because the LRW-trained networks perform better.

2.6 Language: Bangla

In the Bangla language, there are numerous words or symbols that sound remarkably similar. In this paper [24], they investigated the DeepSpeech network for recognizing individual Bengali speech samples. To overcome this, they used ASR which is an algorithm to detect individual characters, words, and sentences. In this paper [24], almost two thousand speech samples which are mostly Bengali real numbers, and a total of 115 unique words are distributed around the data samples. The authors first tried to approximate the phone alignment in individual speech samples using a statistical approach to build their input set. Then, they fed these input samples to a two-layer unidirectional LSTM network and trained their model. But the problem here is recognizing words instead of full sentences. At first, to remove all the noise and distortion, they used Fourier transform on all individual audio samples and extracted highly 13 distinctive nonlinear mel frequency cepstral coefficients (MFCCs) from them. Next, they used Bi-Directional LSTM with CTC loss function for training their data and tested a Bengali real word speech dataset which gives an 8.20% WER and a 3% CER. The limitation of the paper is that it only works on word-level detection, each data sample has eight examples in the dataset and the dataset is very small.

In this paper [25], the authors suggested a convolutional neural network (CNN) architecture for Bangla short speech detection. As the dataset for Bangla is not widely available, they made their own dataset for Bangla short speech commands. Their dataset contains 65,000 samples, considering one second for each word. They took Banglish as their data and then converted them to proper Bangla. They normalized their inputs before training their model with MFCC inputs. They then fed the raw audio data into a similar CNN architecture, which they call the raw model. Prior to training their data, they also pre-trained their model. They used three models for training and testing their dataset. The models are: MFCC, Raw, and Transfer and their training accuracies are respectively 85.44%, 69.08%, and 68.06%. Finally, their testing accuracies are respectively 74.01%, 71.44%, and 73%. In comparison to deep neural networks (DNN), they observed in their research that CNN architecture can lower the error rate on the TIMIT phone recognition dataset by 6-10%. Additionally, they conducted several tests utilizing the limited weight sharing (LWS) and full weight sharing (FWS) schemes and reported that LWS is more efficient since it can recognize feature patterns in many frequency bands. In this research, the main limitation was the dataset was relatively small. Finally, they proposed to expand the words in the dataset in the future so that they can find a better result from the used model MFCC.

In the research [23], they took the dataset from Google Bangla Speech Corpus which has a total duration of 229 hours with 508 native speakers and 50k unique words. Also, test Corpus: 10.6 million unique words and 602.5 million global words, 48.56 million sentences. In this paper[23], they claimed that their CTC-based Bangla ASR end-to-end deep CNN system surpasses the RNN-based DS2 system. For the first time in Bangla ASR research used deep learning techniques to analyze character-wise mistake rates in order to evaluate the corpus's quality. The CNN-based models, which frequently use acoustic characteristics in voice recognition and speaker identification, have been built using MFCC features. They used CTC-CNN and Deep Speech 2 models to train their data. The Deep Speech 2 model is an RNN -CTC-based model. The word error rate in the CNN-CTC model is 36.12% in the Deep Speech 2 model is 40.91%. The advantages of their model are they have a very large dataset, the error rate gets 10% when the characters are found more than 100k times, and characters that are used frequently and have better performance. However, there are some limitations too. Characters found less than 1k times, found 50% error. The characters found less than 1k times, found 50% error. The letter 'R' arrived only 7 times. In this situation, the DS2 model gets 71.4% accuracy. On the other hand, the CNN model could not recognize the word at all. The same pronounced words like 2% cannot be differentiated by this model. The 10 Bangla digits are not very often used in this dataset. So, it has less accuracy than others. In the future, they will work with a large high-order N-gram model to improve Bangla LVCSR.

Chapter 3

Working Plan

Our study seeks to evaluate the effectiveness of the latest deep learning algorithms on the Bangla corpus and develop our own deep learning model appropriate for lip-reading Bangla. For this, at first, we need a Bangla corpus for lip-reading. As there is no known Bangla corpus for lip-reading, we will collect some Bangla videos suitable for our task and make our corpus. Secondly, we will use preprocessing to extract the features for our task. After that, we will split our dataset into train and test data. To train our data, we will use some of the models with the highest success used in lip-reading in other languages, in our own dataset. If those architectures give an accuracy of less than 90, then will try to build a unique model which might provide better accuracy for lip-reading in Bangla.

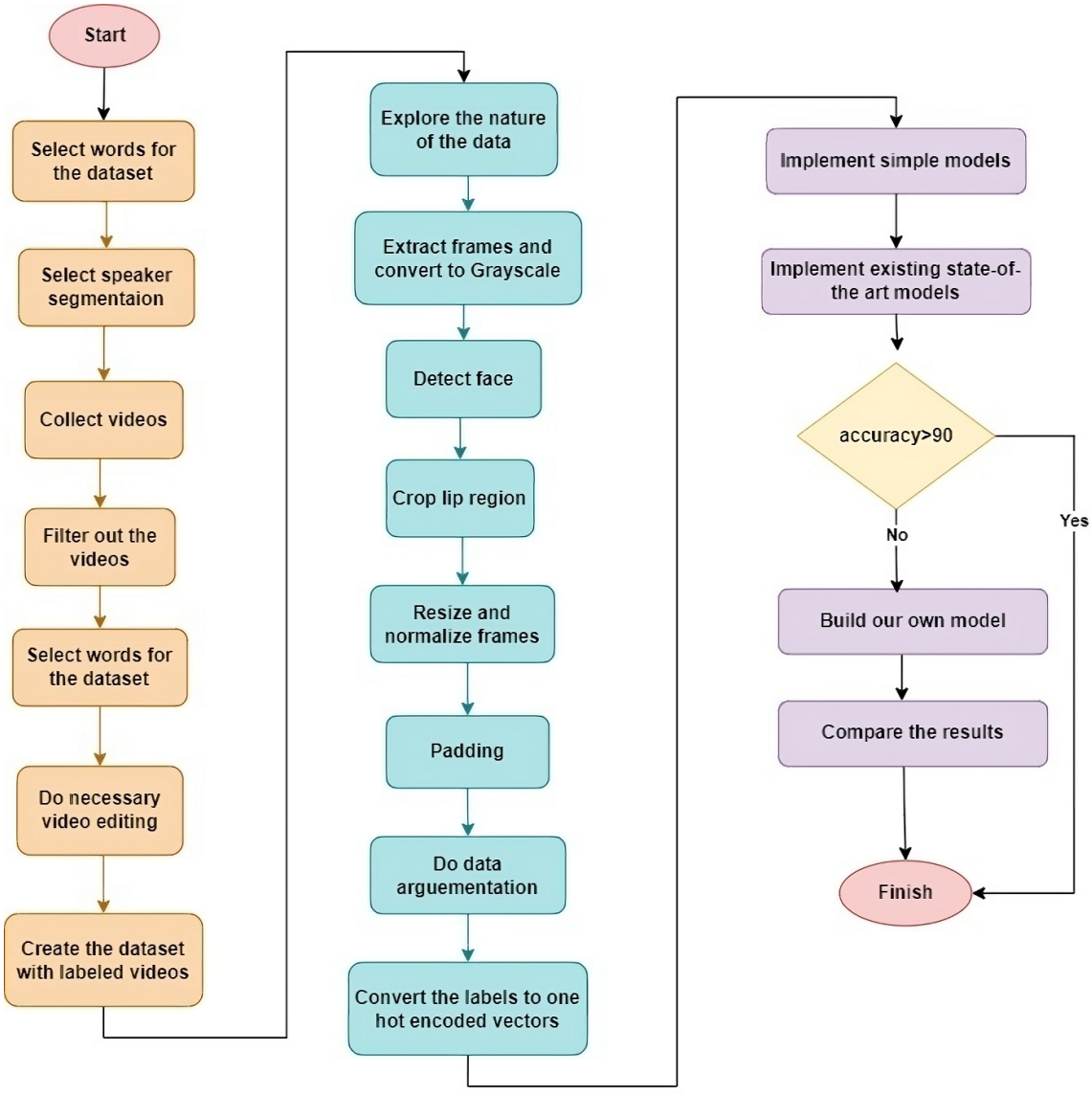

Figure 3.1: Working Plan

Chapter 4

Description of the Dataset

As there was no available video corpus for Bangla language lip reading, we created our own corpus.

4.1 Data collection:

The process of collecting video data from numerous individuals recording themselves speaking specific words typically involves several key steps. Firstly, a well-defined set of words or phrases is chosen for the participants to speak. Next, a diverse group of participants is recruited, ensuring variability in age, gender, and other relevant factors to capture a wide range of linguistic and vocal characteristics. For our data collection, we have selected 25 Bangla words and these words were pronounced by 35 different speakers. The words are:

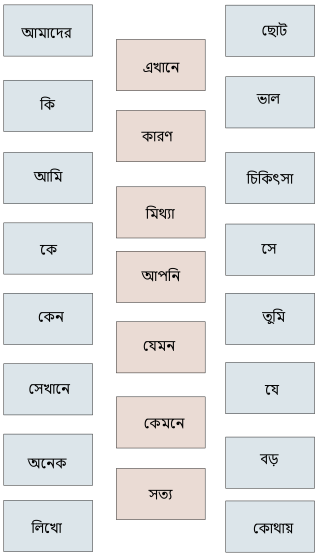

Figure 4.1: Selected words for dataset

Some of these words have the same lip movement like: “Je”, “She” and “Ke”. To collect speakers pronouncing these 25 words is difficult. As in Bangladesh, there are numerous accents and pronunciations. Thus, we have selected the people of Dhaka as our target group of speakers. However, when we were recording these videos we saw that even educated people born and raised in Dhaka pronounces the same word differently. The word “” is pronounced “amra” and “amora” by different speakers. This makes our prediction extremely hard.

Participants were provided with clear instructions and guidelines for recording their videos, which often included technical specifications for video quality. They may use their own recording devices or be provided with standardized equipment. Once the recordings are collected, they undergo a thorough quality control process, where any unusable or subpar videos are filtered out. This step often involves checking for issues like poor lighting. The retained videos are then organized, labeled, and stored in a structured database for analysis. This dataset can serve various purposes, including research in linguistics, voice recognition technology, and accent analysis, among others. Privacy considerations are essential, and participants’ consent and data protection measures must be in place to ensure ethical and legal compliance throughout the data collection process.

4.2 Creating Dataset:

Once the video files are obtained from the speakers, they are meticulously sorted and labeled with spoken content. These annotations help categorize and identify the data efficiently.

Figure 4.2: Visual of dataset

Following this, the collected data is typically converted into a CSV (Comma-Separated Values) format, a commonly used file type for structured datasets. Each row in the CSV corresponds to a specific audio sample, and columns contain relevant information, such as the video file’s path.

4.3 Exploratory Data Analysis:

In the context of our study, data visualization was vital in giving a thorough picture of the hierarchical organization of our dataset. To illustrate the distribution of movies among 25 different folders shown in Figure 4.2, each with a different Bengali word as its name, we specifically used a bar plot. This visualization clearly illustrated how our dataset was distributed, showing that each folder had exactly 35 videos. Bengali terms were used to label the folders, which not only enriched our dataset culturally but also demonstrated our dedication to maintaining linguistic variety in data gathering and annotation. This visualization acts as a key building block for our research, assisting with the first investigation and comprehension of the structure of our dataset, which is fundamental to subsequent analyses and insights drawn from this unique collection of videos.

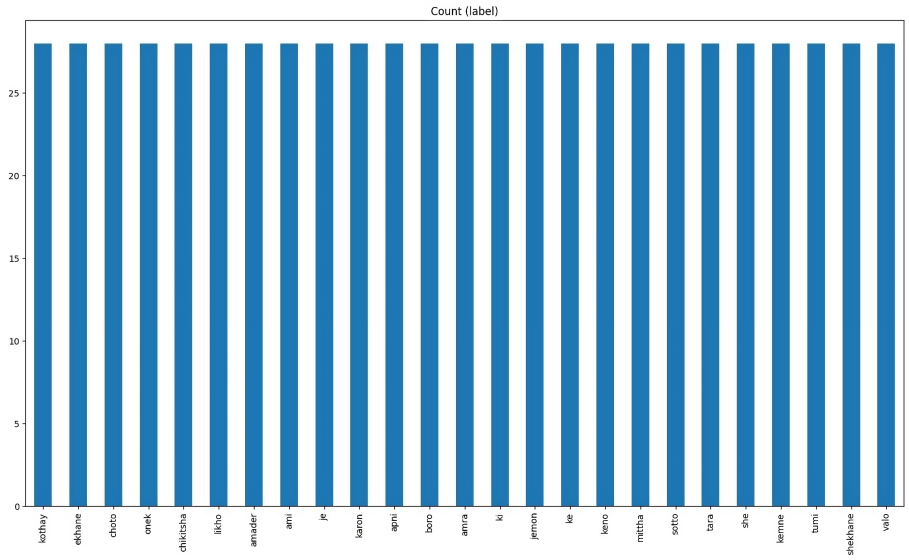

Figure 4.3: Number of videos

Here in Figure 4.3 we can see that all the folders are equally distributed with 35 videos. A sample has been shown in our dataset in Figure 4.4. There we can see a bunch of pictures of different people which are a particular frame from their videos. Also, those videos were in 4 different resolutions and the average length of those videos is 2 seconds shown in Figure 4.5.

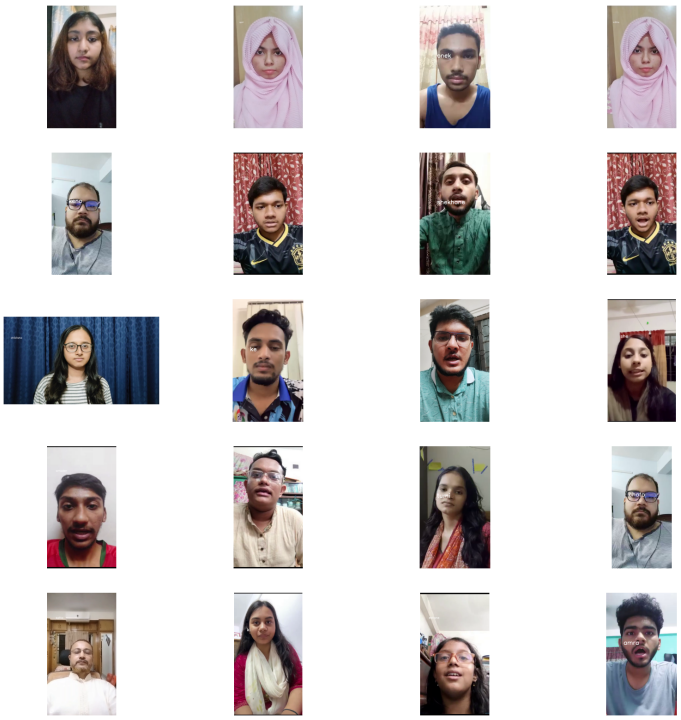

Figure 4.4: Sample frames of different videos

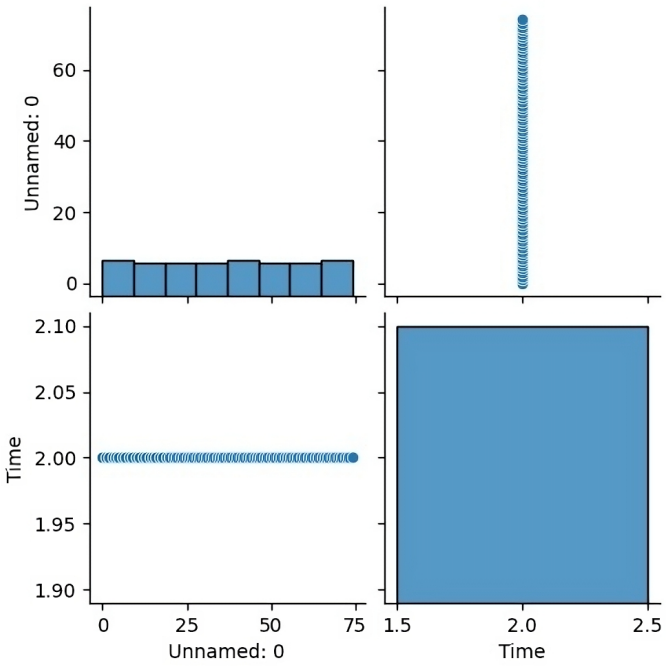

Figure 4.5: Average length of videos

Chapter 5

Pre-processing

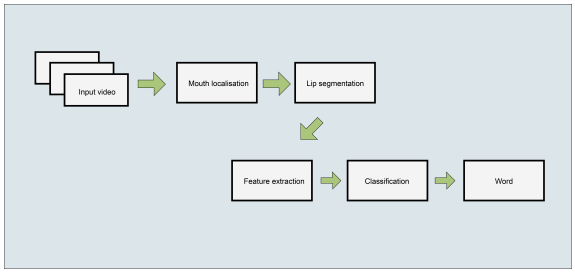

Figure 5.1: Lip-reading processs

5.1 Video editing

When collecting videos from speakers, it's not uncommon to encounter variations in lighting conditions and video file sizes. To address these challenges and ensure the quality and consistency of the dataset, video editing becomes a necessary part of the data preparation process. Videos captured under poor lighting conditions can result in dark, grainy footage that hinders visual clarity. In such cases, video editing software can be employed to adjust brightness, contrast, and color balance to enhance the overall visual quality. This ensures that all videos in the dataset are visually consistent and suitable for analysis or presentation.
However, we used Python by utilizing libraries such as OpenCV and NumPy to manipulate the pixel values in each frame. Moreover, to manage large video file sizes, we trim or cut unnecessary segments from the videos and remove any parts that do not contain relevant content, such as long pauses or redundant sections.

5.2 Extract Frames and Grayscale

Lipreading involves recognizing spoken words and phrases by analyzing the movements and shapes of the speaker's lips over time. Extracting frames allows us to

break down the video into individual moments or time steps, where each frame represents a distinct visual configuration of the lips. This temporal segmentation is vital for capturing the dynamics of speech, phonemes, and mouth movements.
To convert a video into frames, we used a video processing library such as OpenCV in Python. First, we open the video file using the library's video capture function. Then, we read each frame sequentially from the video using a loop. Once we have a frame, we save it as an image file. The number of frames per second (fps) in the video determines the frame rate, which can be used to control how many frames we extract per second. By iterating through all the frames in this manner, we can effectively convert the video into a sequence of individual image frames. In our case, we took a constant number of 30 frames for each video.
Along with converting videos into frames, we converted our videos into grayscale using the 'cv2.COLOR_BGR2GRAY'. Grayscale images contain only intensity information, as opposed to color images that contain red, green, and blue (RGB) channels. By converting to grayscale, we simplify the data and reduce computational complexity, making it easier and faster to detect faces.

5.3 Face Detection & Crop Lip Region

Lip reading focuses on the lip movement of a human face which requires only the lip region. Other than lip region - face, eyes, and expression are not required for lip reading. We removed all the unnecessary regions from the frames and cropped the lip region from the image. For this, we first detected the face and then cropped the lips.

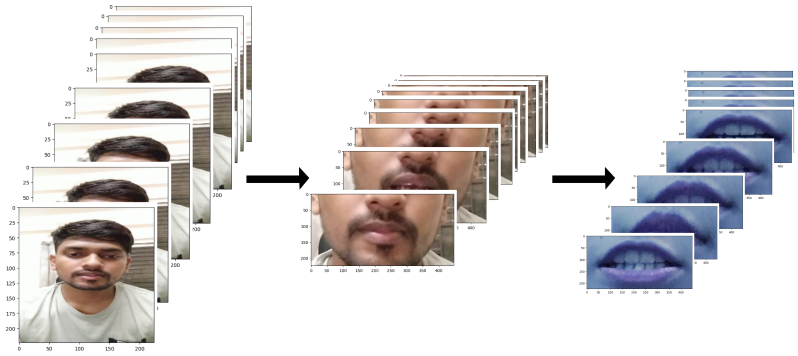

Figure 5.2: Lip segmentation

It begins by using the 'dlib' library to initialize a face landmark predictor and a frontal face detector. The face detector, 'face_detector', identifies the location of faces in the input frame.
Once a face is detected, the landmark predictor, 'landmark_predictor', is employed to obtain the facial landmarks, which are specific points on the face, including those

around the lips. In this case, points 48 to 59 correspond to the lip landmarks. These lip landmarks' coordinates are extracted and used to determine the position and size of the lip region within the face.
To crop the lip region effectively, we code to scale it inward by a factor specified as 'lip_region_scale'. This scaling factor allows adjustment of the size of the extracted lip region. The final 'lip_region' is created by using the computed coordinates (lip_x, lip_y) as the top-left corner and dimensions (lip_width, lip_height) for the region of interest within the face. This 'lip_region' contains the lips of the detected face.

5.4 Resize & Normalize Frames

The videos we have collected from the speaker are from different angles from the camera. Some have uplifted the camera, and some have captured a little from the side. As a result, the lip regions have different sizes in every video or in every frame. Also, every speaker was recorded from a different distance from the camera. This is also a big reason for the size of the frames not to be equal on everything. The cropped frames that have been collected are not equal in every frame or for every speaker. To solve this, all the images have been resized to a fixed height and width of 200,100.
When working with images, it's common to ensure that pixel values fall within a certain range so we used normalization. Normalization typically involves scaling pixel values so that they lie between 0 and 1. By dividing each pixel value for every frame in 255, we are rescaling the pixel values so that they are in the range [0, 1]. This is useful because many image processing and machine learning algorithms work better when pixel values are within this range.

5.5 Padding

Working with the videos of the speakers, every speaker has recorded their videos from different devices that have different frame rates. Having a different number of frames in the dataset can affect the model's accuracy. It's important to maintain consistency in frame counts for training and testing to achieve more reliable and accurate results. Thus, we have set 30-frame standards for our video inputs.
For this, we count the number of frames in each video. If the frame length is more than 30, then it calculates how many frames should be added at regular intervals to create sequences of the desired length then it will add frames at least one frame at a time to maintain consistency. However, if the number of frames for a video is less than 30, it pads with duplicate frames. At first, it calculates the number of frames needed to pad on both sides of the frames to reach the desired length. It then

adds duplicate frames to the end and to the beginning to achieve the padding. The purpose of this padding is typically to ensure that all sequences or frames have the same length, which can be important when working with neural networks or other machine learning models that require fixed input sizes.

5.6 Data Augmentation

For complicated tasks like lip reading, we need an enormous amount of data. Thus we used data augmentation to create variations of the original video frames to enhance the diversity of the training dataset.

When data augmentation is enabled, a data augmentation generator 'datagen' is created specifically for grayscale images. This generator is configured with various image transformation parameters, and we set the rotation range of 20 degrees, horizontal and vertical shifting ranges of 20% of the image size, and the option to horizontally flip images. When applied, these transformations introduce controlled variations to the grayscale frames, simulating different viewing angles, positions, and orientations of the same content. The augmented frames are then added to the original frames, effectively increasing the number of training examples with slightly modified versions of the original frames.

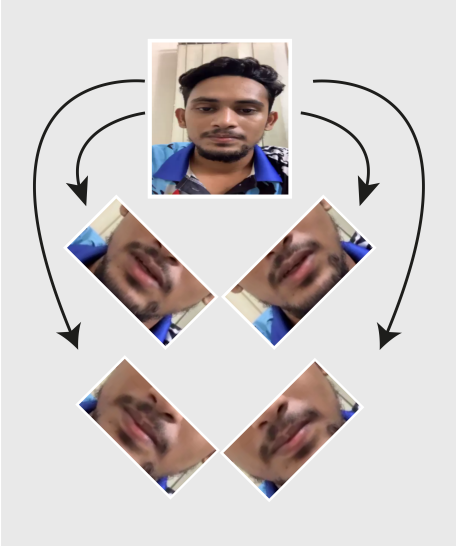

Figure 5.3: Rotated frame of cropped lip clip

5.7 Convert the categorical values

In our dataset, the labels were the 25 words we selected for our classification. These labels were in categorical values and we converted categorical labels to one-hot encoded labels to make them suitable for training, ensuring compatibility, distinct class representation, and proper loss calculation.

Chapter 6

Description of the Model

6.1 Dataset Implementation

Our dataset is a Bangla video corpus of different speakers pronouncing Bangla words. Our target is to predict what a person is saying based only on their lip movement. It is a video classification task. For this task, a CNN-RNN architecture is needed due to the multifaceted nature of video data. To implement this, we try our dataset on two state-of-the-art models that we found during our literature review and analyze its results.

6.2 CNN-RNN architecture

6.2.1 Definition

A CNN-RNN architecture is a neural network model that combines Convolutional Neural Networks (CNNs) for spatial feature extraction from video frames and Recurrent Neural Networks (RNNs) for modeling temporal dependencies in video sequences. It is used for video classification tasks to effectively capture both static and dynamic information in videos. This fusion of spatial and temporal processing is crucial for tasks such as action recognition, video summarization, and scene understanding, where capturing both static and dynamic features is essential for accurate classification.

6.2.2 Working Procedure

A Convolutional Neural Network (CNN) is a type of deep neural network specifically designed for processing grid-like data, such as images and videos. It employs a series of layers, including convolutional layers, pooling layers, and fully connected layers. In video classification, CNNs are used to extract spatial features from individual

frames or video frames. These convolutional layers apply filters (kernels) to detect patterns, objects, and textures within each frame. Pooling layers downsample the feature maps, reducing computational complexity, while fully connected layers at the end of the network interpret the extracted features and make predictions based on them. CNNs are adept at recognizing visual patterns and serve as the initial step in video classification, capturing static information from each frame.

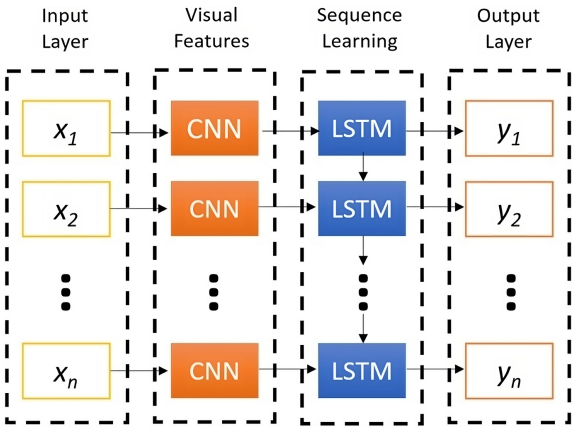

Figure 6.1: CNN-RNN architecture

To perform video classification, CNNs are typically combined with Recurrent Neural Networks (RNNs) or other temporal modeling techniques. The CNN extracts spatial features from each frame, and the sequence of these features is then fed into an RNN. The RNN, with its ability to maintain temporal context and capture dependencies over time, processes these feature sequences to understand the temporal evolution of the video. This combination enables the CNN to handle the spatial aspects of video frames, while the RNN manages the temporal aspect, making it possible to classify videos based on both their static and dynamic content, such as recognizing actions or events within the video.

6.3 Model: LipNet

LipNet [11] is one of the successful models for English lipreading that uses a GRID [7] corpus. Here we tried to implement this architecture on our Bangla corpus. Our model takes as input video data with dimensions (60, 100, 200, 1), which represents videos with 60 frames, each frame having a size of 100x200 pixels in grayscale. Here's a brief explanation of the model:

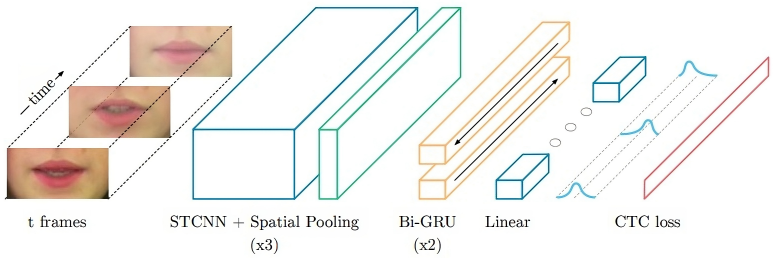

Figure 6.2: LipNetarchitecture

1. Convolutional Layers: The model starts with three 3D convolutional layers, each with different numbers of filters (128, 64, and 30). These layers are used to extract spatial features from video frames. The 'relu' activation function is applied after each convolutional layer to introduce non-linearity, and max-pooling layers are used to downsample the feature maps.

2. Flattening: After the convolutional layers, the model uses the 'TimeDistributed' layer combined with 'Flatten()' to reshape the output from the convolutional layers. This step prepares the data for sequential processing while maintaining the temporal structure of the video frames.

3. Bidirectional GRU Layers: The model employs two layers of Bidirectional Gated Recurrent Units (GRU) for temporal modeling. Each GRU layer has 128 units and returns sequences (rather than just the final output). Bidirectional GRUs are capable of capturing temporal dependencies in both forward and backward directions, allowing them to understand the temporal dynamics in the video.

4. Dense Layer: The final layer is a dense layer with 25 units, which corresponds to the number of classes for video classification. It uses the 'softmax' activation function, enabling the model to output class probabilities and make predictions.

5. Categorical Crossentropy: Here instead of CTC loss, we use categorical crossentropy. This loss function measures the dissimilarity between the predicted class probabilities and the true class labels, encouraging the model to assign high probabilities to the correct video classes while penalizing it for incorrect predictions. It is commonly used for multi-class classification tasks like video classification and helps train the model to make accurate class predictions during training.

Overall, this architecture combines convolutional layers for spatial feature extraction, GRU layers for temporal modeling, and a softmax dense layer for classification.

It is suitable for video classification tasks, where both spatial and temporal information need to be considered for accurate classification.

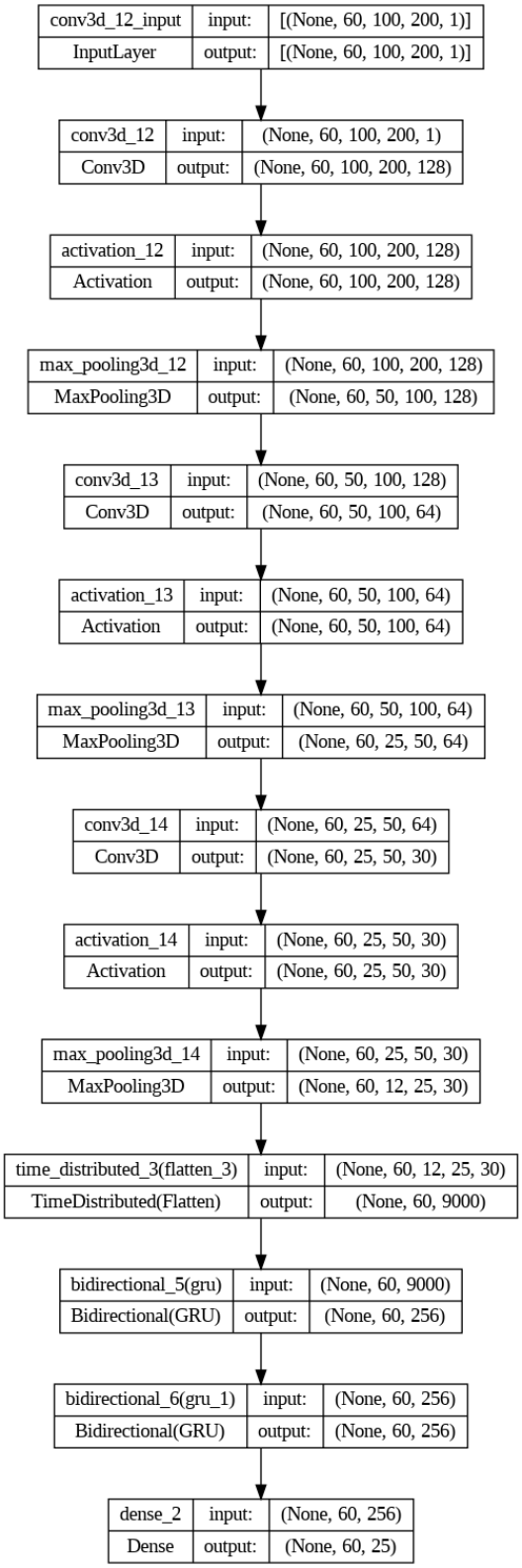

Figure 6.3: Implemented Model Details

Chapter 7

Analysis

7.1 Model Result

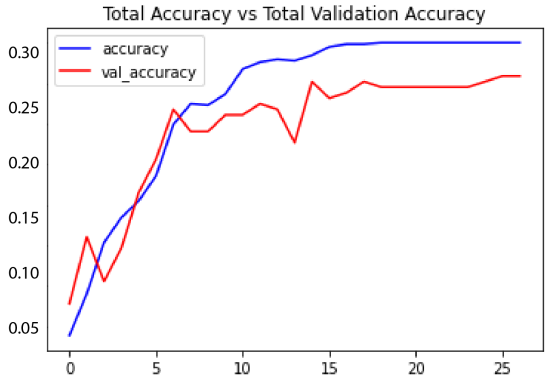

Figure 7.1: Total accuracy vs. Total validation accuracy graph for classification

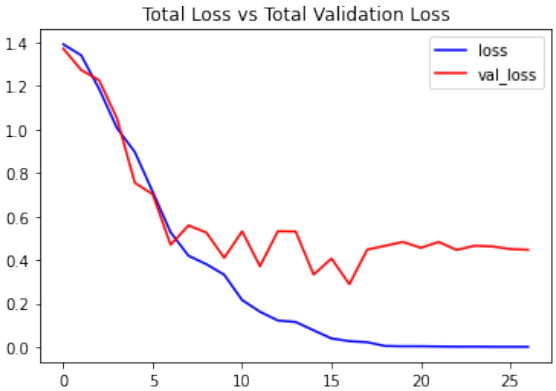

Figure 7.2: Total loss vs. Total validation loss graph for classification

From the proposed model two different graphs have been generated. Figure 7.1 describes the accuracy and validation accuracy of the proposed model. The image shows a graph with two lines, one red and one blue rising in an upward direction.

There are several points along each line that indicate changes in their respective directions. Training our model for 25 epochs, an intriguing observation emerged, wherein the validation accuracy consistently lagged behind the total accuracy. This phenomenon warrants careful consideration as it carries implications for the model's generalization and performance on unseen data. On the other hand, Figure 7.2 represents the total loss vs. total validation loss graph again with two curves. Here, validation loss was slightly lower in the starting phase, and with the time frame total loss is ultimately lower than the validation loss. for only 25 epochs.

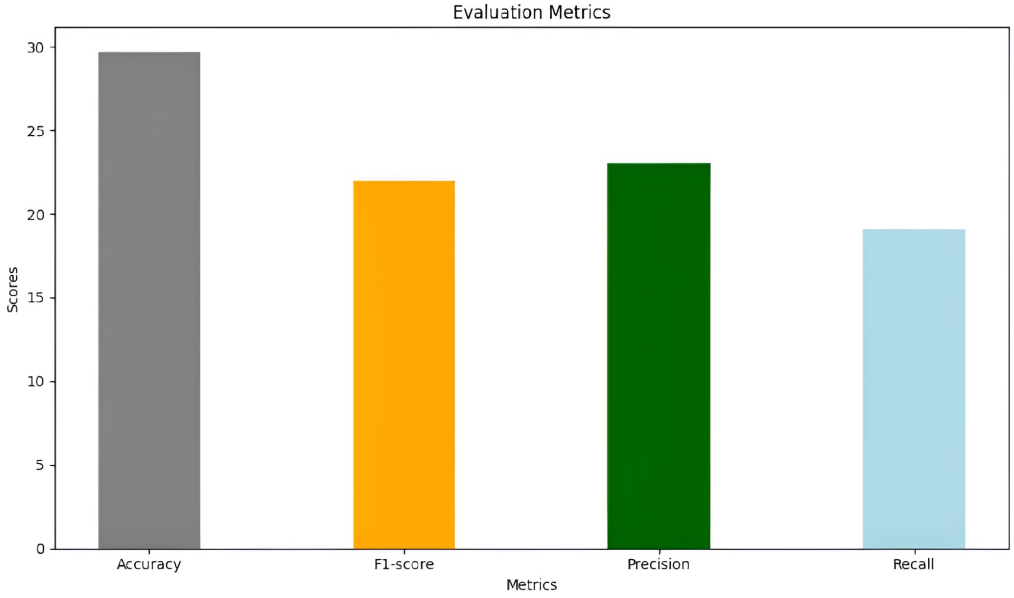

Figure 7.3: Evaluation Metrics

7.2 Future Work

In our research, we have implemented the current existing model. However, the result we got is not satisfactory which indicates that this model might not be suitable for Bangla language lipreading. Hence, in our future work, we will try to build our own model and keep increasing the Bangla video corpus for greater accuracy.

Chapter 8

Conclusion

In conclusion, lipreading is the skill of understanding a speaker's words solely from the movement of their lips which is crucial in many different disciplines like accessibility, noise-sensitive communication, human-computer interaction, multimodal communication, security, and a great deal more. Despite the fact that the English language has been the subject of numerous studies in this area, the Bangla language has not yet been the subject of any studies. CNN, Bi-LSTM, Bi-GRU, and CTC were frequently employed for English lipreading. However, after reviewing some studies, we discovered that the lipreading models utilized for English were inadequate for other languages. Therefore, it is imperative that we carry out research on Bangla lip-reading. However, there is not yet a corpus of Bangla that can be used for lipreading. In our study, we'll build our own corpus and train cutting-edge lip-reading methods for English to test how well they function on Bangla datasets. Finally, we'll attempt to create a custom model that works with Bangla lip reading.

Bibliography

[1] D. Easton and M. Basala, "Perceptual dominance during lipreading," *Perception & Psychophysics*, vol. 32, no. 6, pp. 562–570, 1982.

[2] J. Goldschen, O. N. Garcia, and E. D. Petajan, "Continuous automatic speech recognition by lipreading," in *Motion-Based Recognition*, Springer, 1997, pp. 321–343.

[3] G. Neti, J. Potamianos, J. Luettin, *et al.*, "Audio visual speech recognition," IDIAP, Tech. Rep., 2000.

[4] A. Rix, J. Beerends, M. Hollier, and A. Hekstra, "Perceptual evaluation of speech quality (pesq), an objective method for end-to-end speech quality assessment of narrowband telephone networks and speech codecs," ITU-T Recommendation, Tech. Rep., 2001.

[5] T. F. Matthews, J. A. Cootes, J. Bangham, S. Cox, and R. Harvey, "Extraction of visual features for lipreading," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 198–213, 2002.

[6] M. Elhilali, T. Chi, and S. Shamma, "A spectrotemporal modulation index (stmi) for assessment of speech intelligibility," *Speech communication*, vol. 41, no. 2, pp. 331–348, 2003.

[7] J. Cooke, S. Barker, D. Cunningham, and X. Shao, "An audio-visual corpus for speech perception and automatic speech recognition," *The Journal of the Acoustical Society of America*, vol. 120, no. 5, pp. 2421–2424, 2006.

[8] M. Zhao, M. Barnard, and M. Pietikainen, "Lipreading with local spatiotemporal descriptors," *IEEE Transactions on Multimedia*, vol. 11, no. 7, pp. 1254–1265, 2009.

[9] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: The first facial landmark localization challenge," in *IEEE International Conference on Computer Vision Workshops*, 2013, pp. 397–403.

[10] G. Zhou, G. Zhao, X. Hong, and M. Pietikainen, "A review of recent advances in visual speech decoding," *Image and Vision Computing*, vol. 32, no. 9, pp. 590–605, 2014.

[11] Y. M. Assael, "Lipnet: End-to-end sentence-level lipreading," *arXiv preprint arXiv:1611.01599*, 2016.

[12] S. Gergen, S. Zeiler, A. H. Abdelaziz, R. Nickel, and D. Kolossa, "Dynamic stream weighting for turbodecoding-based audiovisual asr," in *Interspeech*, 2016, pp. 2135–2139.

[13] H. Akbari, "Lip2audspec: Speech reconstruction from silent lip movements," *arXiv preprint arXiv:1710.09798*, 2017.

[14] A. Ephrat and S. Peleg, "Vid2speech: Speech reconstruction from silent video," *arXiv preprint arXiv:1701.00495*, 2017.

[15] T. Stafylakis and G. Tzimiropoulos, "Combining residual networks with lstms for lipreading," in *conference of the international speech communication association*, 2017, pp. 3652–3656.

[16] T. Afrouas, J. Chung, and A. Zisserman, "Deep lip reading: A comparison of models and an online application," in *ArXiv*, 2018.

[17] "Bangla short speech commands recognition using convolutional neural networks," *IEEE Conference Publication — IEEE Xplore*, 2018.

[18] M. Faisal, "Deep learning for lip reading using audio-visual information for urdu language," *arXiv preprint arXiv:1802.05521*, 2018.

[19] L. McQuillan, "Is lip-reading the secret to security?" *Biometric Technol Today*, vol. 2019, pp. 5–7, 2019.

[20] B. Purkaystha, M. Nahid, and M. Islam, "End-to-end bengali speech recognition using deepspeech," *ResearchGate*, 2019.

[21] X. Chen, J. Du, and H. Zhang, "Lipreading with densenet and resbi-lstm," *Signal, Image and Video Processing*, vol. 14, no. 5, pp. 981–989, 2020.

[22] N. Ali, M. Abdulmunem, and A. Ali, "Constructed model for micro-content recognition in lip reading based deep learning," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 5, pp. 2557–2565, 2021.

[23] "Hlr-net: A hybrid lip-reading model based on deep convolutional neural networks," 2021.

[24] A. Samin, M. Kobir, S. Kibria, and M. Rahman, "Deep learning based large vocabulary continuous speech recognition of an under-resourced language bangladeshi bangla," *Acoustical Science and Technology*, vol. 42, no. 5, pp. 252–260, 2021.

[25] B. Soundarya, R. Krishnaraj, and S. Mythili, "Visual speech recognition using convolutional neural network," *IOP Conference Series: Materials Science and Engineering*, vol. 1084, no. 1, p. 012 020, 2021.

[26] "A novel method for lip movement detection using deep neural network," *Journal of Scientific & Industrial Research*, vol. 81, no. 06, 2022.

[27] U. Atila and F. D. Sabaz, "Turkish lip-reading using bi-lstm and deep learning models," *Engineering Science and Technology, an International Journal*, vol. 35, p. 101 206, 2022.

[28] M. Miled, M. Messaoud, and A. Bouzid, "Lip reading of words with lip segmentation and deep learning," *Multimedia Tools and Applications*, vol. 82, no. 1, pp. 551–571, 2022.

[29] G. Schwiebert, C. Weber, L. Qu, H. Siqueira, and S. Wermter, "A multimodal german dataset for automatic lip reading systems and transfer learning," Tech. Rep., 2022.