Kaushik Sarkar, Pranab Hazra, Anilesh Dey, Saradindu Panda, Surajit Bari,
Soumen Pal, Arpita Santra, Swati Barui, Sangita Roy, Abhijit Ghosh,
Sandhya Pattanayak, Puspak Pain, Arnima Das, Moupali Roy, Rimpi Datta

Turkish Online Journal of Qualitative Inquiry (TOJQI)
Volume 12, Issue 6, June 2021: 1296-1304

Detection of word boundary in Bengali colloquial speech based on pitch profile and Convolutional Neural Network

Kaushik Sarkar[a], Pranab Hazra[b], Anilesh Dey[c], Saradindu Panda[d], Surajit Bari[e],
Soumen Pal[f], Arpita Santra[g], Swati Barui[h], Sangita Roy[i], Abhijit Ghosh[j],
Sandhya Pattanayak[k], Puspak Pain[l], Arnima Das[m], Moupali Roy[n], Rimpi Datta[o]

[a* ,b,c,d,e,f,g,h,I,j,k,l,m,n,o]ECE Dept, Narula Institute of Technology Kolkata, West Bengal, India

email:[a]kaushik.sarkar@nit.ac.in,[b]pranabhazra2017@nit.ac.in, [c]anileshdey@ieee.org

Abstract

Word segmentation is a decisive part in any speech to text conversion. Works have been done on popular languages for word boundary, especially on English. But a very few work is conducted on Bengali language, especially on colloquial speech. In speech recognition word boundary detection is significant and crucial task. To understand and fixing the problems on efficiently detecting where the words are present in a signal is still challenging. In this paper, we extract the feature of the existence of words based on the pitch profile of a speech. We cluster the pitch profile and Silhouette index is used to select the words boundary with the help of Convolutional Neural Network (CNN)..

Keywords: Signal Processing, word boundary, pitch profile, Convolution Neural Network

1. Introduction

Speech signal consists of the words spoken along with some silence parts comprising random noises. During continuous speaking, the consecutive words have very small pauses between them. Thus, it becomes difficult to separate them within a signal. In this paper, the first approach to detect the word boundary using pitch profile with the help of state phase diagram. Then the Convolutional Neural Network (CNN) is used

Here we have considered some basics of human voice. Human voice can be classified broadly in three categories like Silence (S), Unvoiced (U) and Voiced (V) as shown in Fig.1. It is noted that a word consists of voiced and unvoiced part and silent section indicates the absence of voice. To find the word in a voice signal, we are looking for the voiced and unvoiced part. Our vocal cord generates the voiced part and it has a quasi-periodic nature. The period of such signal is commonly known as pitch. So, our idea is to extract the pitch of a signal to estimate the presence of word. In the next section, we describe the method for pitch detection.

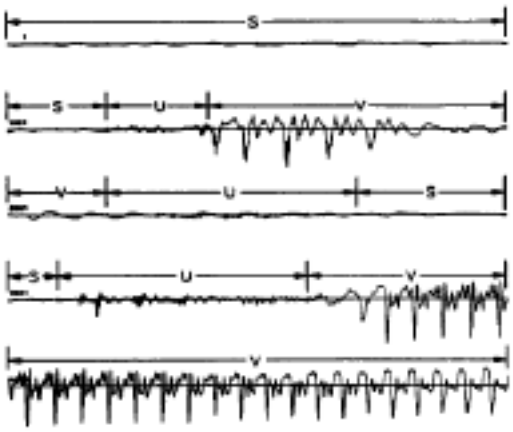


Fig: 1 Position of Silence (S), Unvoiced (U) and Voiced (V)

2.Methodology

For analyzing the speech signal State Phase is used. Let x(n) is a time series then Figure-2 shows the relation between x(i) and x(i+L), where L is the amount of delay. If the relationship between x(i) and x(i+L) is periodic then we get a symmetrical line, but in the Figure-3 the line is not symmetrical but close to symmetrical that means the nature of the signal is quasi periodic. We need to find this spread to identify the periodic and quasi periodic state of a signal.

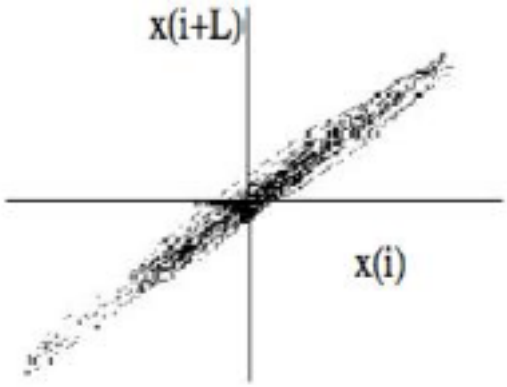


Figure 2: phase difference of 2π

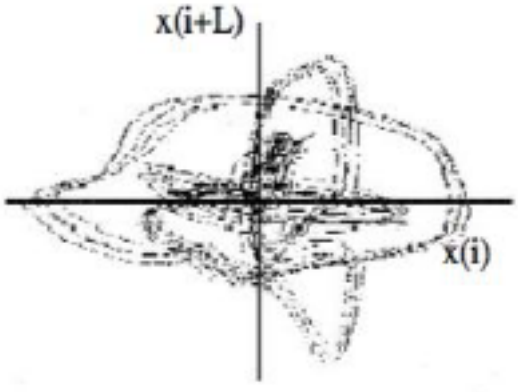


Figure 3: phase difference of $\frac{\pi}{2}$

Figure-4, depicted that the x point is deviated from the straight line with respect to the point p, and the amount of deviation can be measured from equation (1).

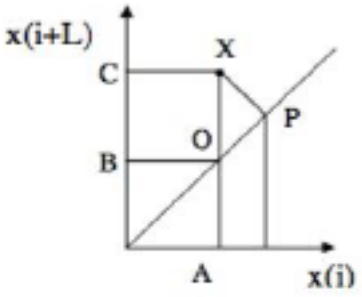


Figure 4: State phase diagram

In our study, we use this deviation as Average Magnitude Difference Function as in equation-

$s = \frac{1}{N} \sum (x(i+L) - x(i))$. -------- (1

*w*here, N=Number of point present in each section, $(i+L) = delay\ point$, L=delay.

The nature of the Deviation vs delay for quasi-periodic, quasi-random and silence has been shown in Figure-5,6,7.

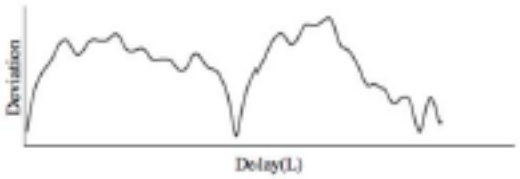


Figure 5: Deviations against delay for quasi-periodic signal

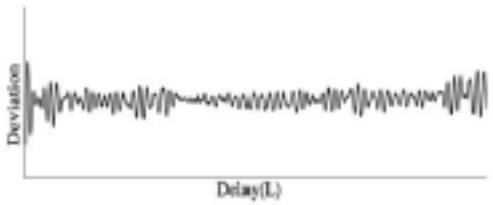


Figure 6: Deviations against delay for quasi-random signal

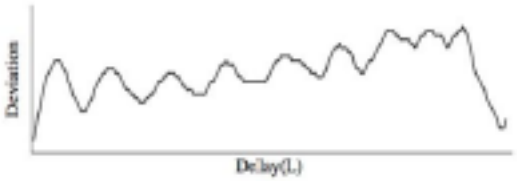


Figure 7: Deviations against delay for silence in the speech signal

3.Pitch detection algorithm:

1. Set, t=20ms window, ~~t1~~ t_1=Fs*t/1000, where Fs=sampling frequency of the signal, ~~t1~~ t_1=how many sample point present for a particular window length (t).
2. Set,l=1,x(i)=1….~~t1~~ t_1,x(i) will continue until the length of x(i) is greater than equal ~~t1~~ t_1.x(i+L).And then using the ~~eqn 2~~ equation 2,we get the deviation for t ms window.
3. Pitch (f) = Fs/data
4. Take the pitch within (75-500) Hz.
5. pitch_norm=Data/max(Data),using this formula normalize the pitch data, and this is use only for visual assessment.

In Figure-8, blue color indicates the speech signal and pitch profile is shown in red color. It is observed that there are some areas marked in black arrow, where there is no pitch, but still our algorithm finds pitch. To

remove those unwanted pitch power profile of a speech has been used. The corresponding power profile is shown in Figure-9.

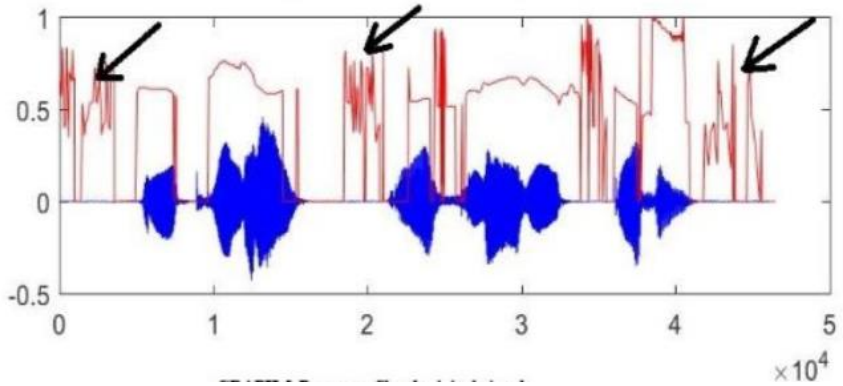


Figure 8: Red color for pitch profile and Blue color for Speech signal

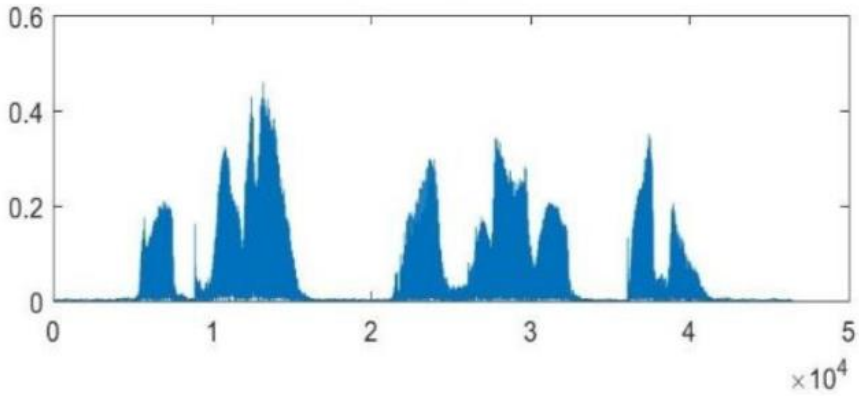


Figure 9: Power profile

We multiply the power profile with the pitch profile to discard the unwanted areas of pitch as shown in Figure-8. The final pitch profile is shown in Figure-10.

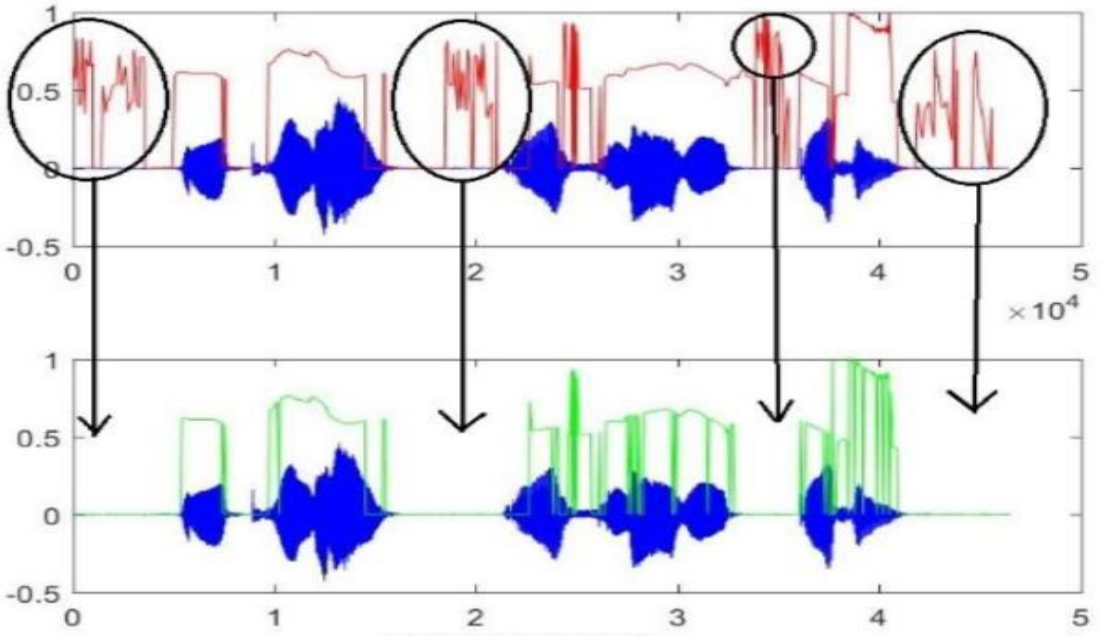


Figure10: Pitch profile (in green) after using power profile

Some designed CNN model deal with the 2D shapes which has shown to outperform in speech recognition and found to work efficiently [4]. CNN model is a type of neural network which is used for visual imagery analysis. It recognizes pattern and based on them it classifies the images. It finds the relation between two nearest pixels and tries to recognize any pattern or higher dimensional features which is present in an image.

In CNN approach, first we have created word profiles of the audio signals. Then we have done some pre-processing to reduce the input and output shape. Finally, Mel spectrogram transformation is done to create the training dataset.

In this method, some transformations on the raw audio signal is done to get the features from the audio signals. Here we have used MFCC matrix as the input to the model. Word profiles are used as the labels for the model.

- For this total work, first we have to trace that raw audio signal's portion where the words are present. We can get the word boundaries by Convolutional Neural Network (CNN)

We will further discuss about these techniques. Before that we will explain few signal processing techniques which we used in this paper.

3.1.Mel Spectrogram

Spectrogram is a powerful visualization tool to represent the signal strength of a signal. It is basically a two-dimensional graph where horizontal axis is the time axis, vertical axis is frequency axis and amplitude is represented by color intensity where the dark color implies lower amplitude and the bright color implies higher amplitude.

In case of a speech signal, we transform the frequency axis into log scale and color (amplitude) axis into Decibels.

The human auditory system interpretation of pitch is not in linear manner [3]. To represent this in a linear scale, the Mel Scale was developed by Stevens, Volkmann and Newman in 1937 by experimenting with human ears interpretations of pitch [2]. This scale is constructed by assigning a perceptual pitch of 1000 Mels to a 1000Hz frequency, 40 dB above the listener's threshold and above 1000Hz the scale shows a logarithmic nature. The formula for Mel scale is-

$$Mel(f) = 2595 \log(1 + \frac{f}{700})$$

Mel scale spectrogram is a spectrogram where the y axis is the Mel scale.

3.2.Mel Frequency Cepstral Coefficients (MFCC)

[3] Speech is the sound which is generated by a human by changing the shape of the vocal tract which includes tongue, teeth etc. The shape of the vocal tract is responsible for the variation of sound. If we can determine the shape accurately, then we can predict which phoneme is being produced. The shape of the vocal tract manifests itself in the envelope of the short time power spectrum. Here MFCC comes to accurately get the envelope from the audio signal.

Mel Frequency Cepstral Coefficients (MFCCs) is a mostly used feature in automatic speech and speaker recognition.

3.3.Word Boundary Detection Using Convolutional Neural Network

We are supposed to create a model which can take an audio signal as input and can tell what are each word's starting and ending position in that signal.

Let us consider the following audio signal.

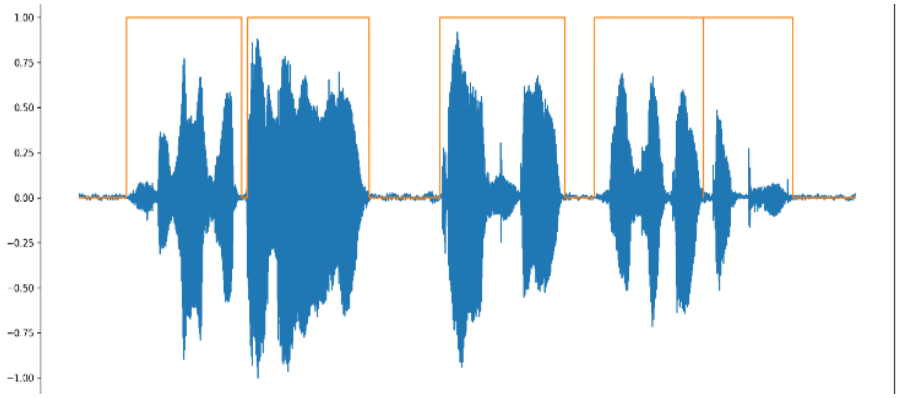


Figure-11: Raw audio waveform with word boundary

Just by looking at this raw audio signal we can't get any clue about where the word breaks and another problem is different audio signals have different lengths. The required model is supposed to work for audio signals of any length. The audio signal shown above has five words.

3.4.Create Word Profile

Here we have used 150 audio signals as our dataset. We have manually taken each word's start time and its duration and saved it as a csv file. Then we have created a profile of words from the csv files.

The csv file looks like this:

name	start	duration
marker 01	0:00.202	0:00.491
marker 02	0:00.718	0:00.517
marker 03	0:01.537	0:00.532

Steps:

```
length = length (audio signal)
word profile = zeros (shape = (length))
for each marker in (no. of markers):
   end time = start time + duration
   word profile [start time: end time] = 1
end for
```

In this way we have created a word profile for the signals which looks like the orange profile shown in figure 11.

Note: The word profile has same length as the audio signal.

Now the word profile holds all the information:

- where the words are
- how many words are there
- every word's starting and ending time or position

Now the main idea is, we're giving audio signal to a model as input and it will give the signal's word profile as output.

But this is difficult because, suppose a normal audio length is 3 sec and if it's sampling frequency is 22050/sec, then we have a length of audio signal of 3 x 22050 = 66150 samples and the length of word profile is also same i.e. 66150 samples and feeding raw audio signal to train a model is not a good idea. So, we have done some pre-processing on the audio signals.

3.5.Data Preprocessing

Wehave divided the signal in 0.5 sec segments. We have done the same thing to the word profile.

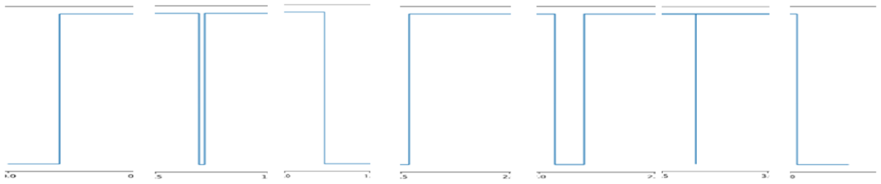


Figure-12: 0.5sec segment of a complete word profile

Now we have 0.5 sec input and 0.5 sec output. If sampling frequency = 22050/sec then input shape = 22050 * 0.5 = 11025 samples and output (word profile) shape = 11025 samples.

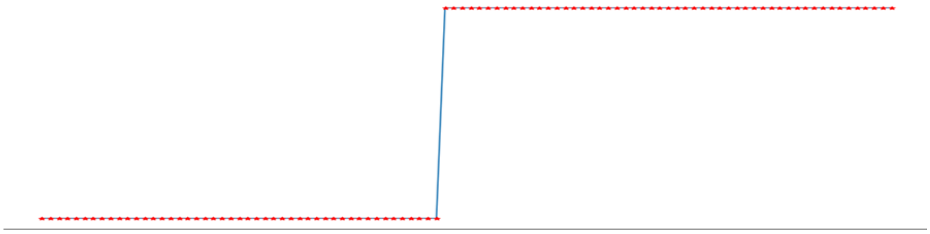
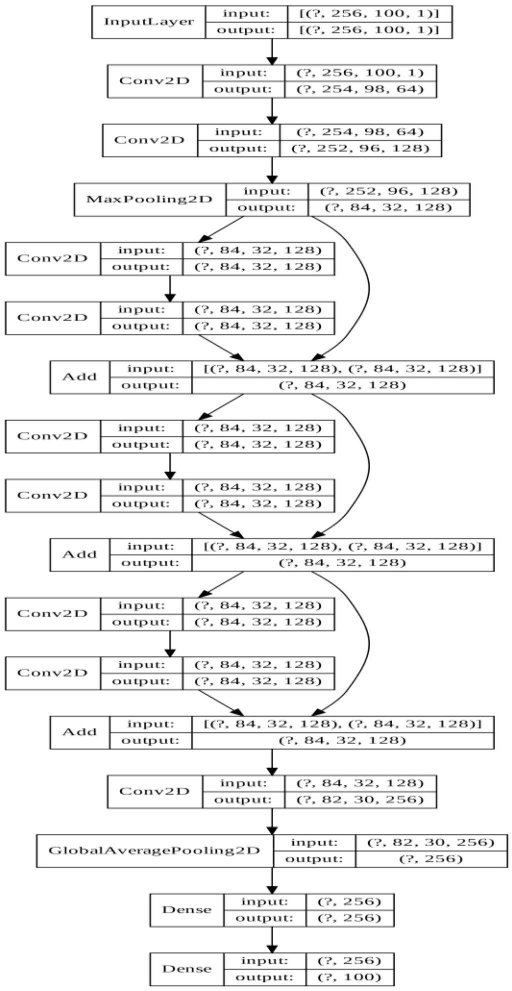


Figure-13: Word profile with 5ms interval points

The red dots in figure 13 are the 5ms interval points. The sequence is 500ms, so we have taken one value for every 5ms interval which is 0 or 1. These are enough values to represent the profile. So now, numbers of points will be 500ms /5ms = 100 samples

Finally, we have 11025 numbers of points for input signal for model and 100 numbers of output points.

3.6.Model Architecture



3.7.Create Training Dataset

Now the model can be trained but there is still one problem left. Training the model with raw audio signal is not a good idea. We have to transform the audio from time domain to any other domain.

The transformation options are: 1) FFT, 2) STFT,3) Spectrogram, 4) Mel Spectrogram, 5) MFCC.

Among these Mel spectrogram gives the best result, so we have performed this transformation.

To get the 0.5 sec Mel spectrogram we need to set the parameters:

window = 0.005sec = 5ms, fs = 22100

We've used python library librosa to get the mel spectrogram.

[2]librosa.feature.melspectrogram(y=audio_samples_array, sr=fs, n_mels=256, fmax=8000, hop_length=int(fs*window), n_fft=1024)

This gives 256x100 matrix for every 0.5 sec. Now we have a [256x100] matrix. This matrix is the input to the model.

4.Result Analysis

4.1.Predict Word Profiles using Convolutional Neural Network approach

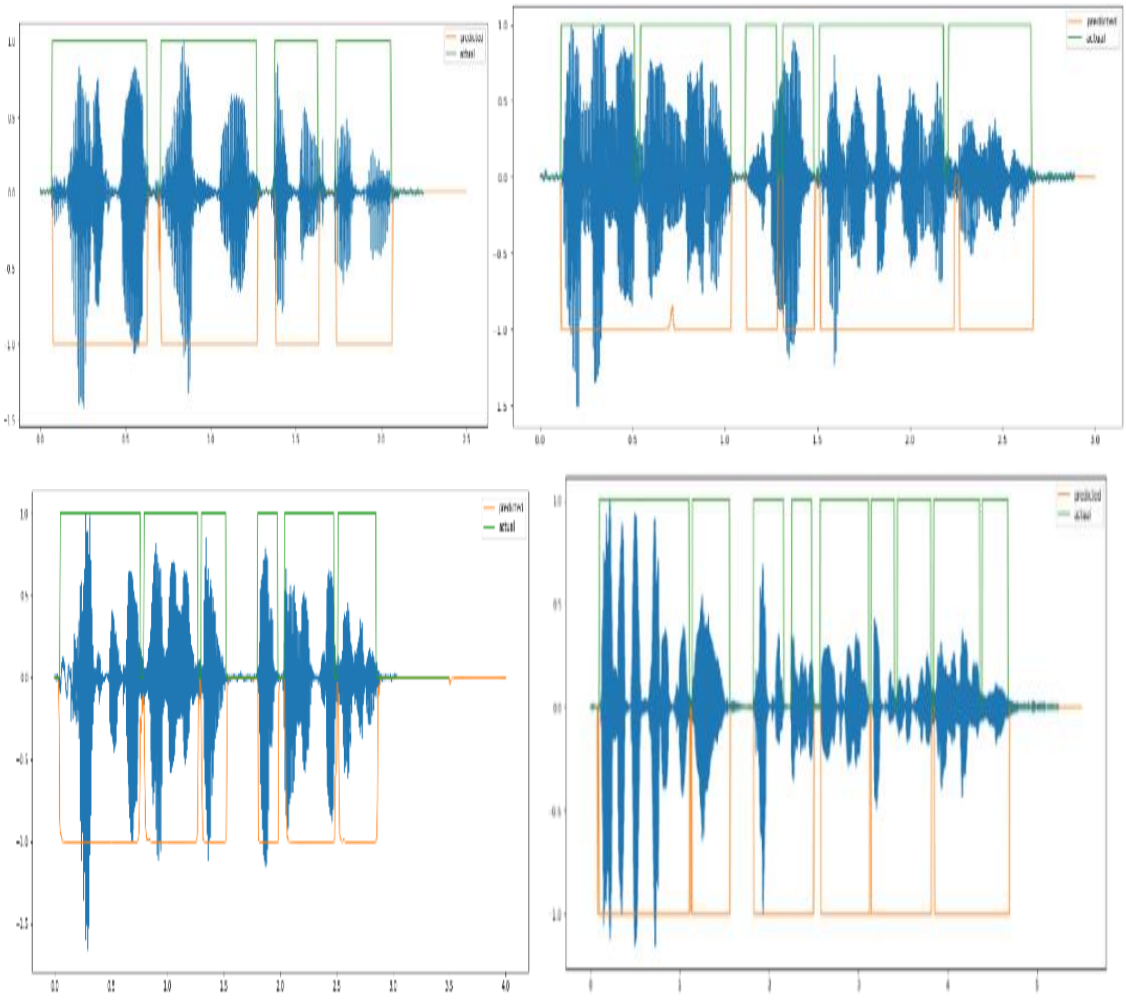


Figure-14: 4 Predicted Word Profiles using Convolutional Neural Network model

From the Results we can conclude that the first three figures in figure-14 the model is able to predict all the word profiles correctly but in last figure first two words are predicted as one word but the rest of the words are predicted correctly. In the final figure the model predicts some individual words together but there is no wrong prediction if we consider all the word boundaries.

5.Evaluation

5.1.Accuracy of Convolutional Neural Network (CNN) Model in Word Boundary Detection

We have to binarize the model's predicted profiles with some threshold value. Different threshold value gives different accuracy. So we have calculated the accuracy with different threshold values and selected the threshold value which gives high accuracy.

The model's prediction is not correct always, sometimes it predicts some individual words together but if we consider only the word boundaries, the output of the model is not wrong. So, to measure the accuracy based on the number of words predicted by the model and actual number of words present in that audio signal, if the model's predicted words and actual number of words are same then we call it as fully accurate and if the model predicts some two words together but predict the other words correctly then we call it as mostly correct. Finally, we have calculated the overall accuracy by adding the two accuracy values.

Table 1. Accuracy table of Convolutional Neural Network (CNN) model with different threshold values

Threshold	Fully Accurate	Mostly Accurate	Overall Accuracy
0.5	0.171	0.4	0.571
0.6	0.214	0.471	0.685
0.7	0.200	0.514	0.714
0.8	0.341	0.4	0.741
0.9	0.285	0.471	0.756

From the table we can see that the threshold value of 0.9 gives highest accuracy so we take the threshold value as 0.9.

6.Conclusion

In this paper, we have used two methods to get the word boundaries from an audio signal. The CNN method output profile is noisy but it produces good output for unseen data. If we apply some threshold to remove the noise, it will show a perfect output. Sometimes it failed to find some intermediate words, but in most of the cases it produces good output.

In case of pitch profile, its output profiles are nearly accurate and noise free but it doesn't give good results for all data. In case of unseen data, sometimes both the model failed to recognize the intermediate word, or predicting two words together.

CNN model's performance is good but lot of computation is required for a single audio file prediction, but pitch profile model doesn't require that much computation. its prediction is quite fast. This is an easier method to find the word boundary.

References

[1] Acoustics of Bangla Speech Sounds, Asoke Kumar Datta, Springer Publications

[2] Librosa: A Python Audio Library, https://medium.com/@patrickbfuller/librosa-a-python-audio-libary-60014eeaccfb

[3] Mel Frequency Cepstral Coefficient (MFCC) tutorial, http://practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficients-mfccs/

[4] Yann LeCun, Yoshua Bengio & Geoffrey Hinton, "Deep learning" 436, NATURE, VOL 521, 2015, doi:10.1038/nature14539