Preprocessing of Continuous Bengali Speech For Feature Extraction

Md. Mehedi Hasan
Department of Computer Science and Engineering
Daffodil International University
Dhaka, Bangladesh
mehedi15-9804@diu.edu.bd

Hasmot Ali
Department of Computer Science and Engineering
Daffodil International University
Dhaka, Bangladesh
hasmot15-9632@diu.edu.bd

Md. Fahad hossain
Department of Computer Science and Engineering
Daffodil International University
Dhaka, Bangladesh
fahad15-9600@diu.edu.bd

Sheikh Abujar
Department of Computer Science and Engineering
Daffodil International University
Dhaka, Bangladesh
sheikh.cse@diu.edu.bd

Abstract— **As Voice is the most suitable form of communication, voice-based applications are playing a vital role in modern technology for the last few decades. It is not only the trend of modern and efficient technology but also a new shift of information and technology paradigm. Several research works have been completed on voice-based applications because it has more practical application than any other form of communication. For almost every application, voice signals need to be preprocessed before using it as the input signal. Preprocessing of any data improves the performance of applications. A preprocessing method for feature extraction of continuous Bangla voice has been proposed in this paper. In this method the signal is preprocessed in several steps. In the first step, the noise has been reduced from the signal. Then after balancing the frequency applying pre-emphasis, framing is applied which splits the whole signal into some frames. After that a hamming window and normalization are applied to improve the spectrum and SNR of the preprocessed signal. By following these steps, a clear voice signal, free from noise and unnecessary frequencies has been retrieved.**

Keywords— Speech Preprocessing, Noise Cancellation, Feature Extraction, Fast Fourier Transform

I. Introduction

The human voice is the most suitable way of communication having perfectly structured, multiple level regularizes properties but only a few fragments of these properties become significant units, which is defined as phonemes[1-2]. In the field of Machine Learning (ML) preprocessing is the savior of many random and useful data when we have to work with data that is not ready for training or impossible to encode as machine-readable form. For working with voice or speech a several types of preprocessing research is performed from the last decade. Feature extraction is of the most important preprocessing as a lot of speech application depends on a different feature of voice. The unique feature of voice proposed a lot of applications such as Speech Recognition, Emotion Recognition, Phonetic Discrimination, Classification for Classroom Speech Intelligibility, Gender Detection, Age Recognition from voice, and many more [3]. Extraction of real and unique feature from a speech is a challenging work as a real voice contain many unwanted entities. A user delivery speech contains three kinds of entities as follows The Voice Speech, The Unvoiced Speech, and Silence[4]. Therefore, the unvoiced speech and silent part of speech are unnecessary and useless for process and extraction and this is the most challenging part to separate the useful voiced data for performing feature extraction. This is why the accuracy and use of speech feature-based application is comparatively less and limited feature. For extraction of the real feature of voice, the first step is to removing the unvoiced and silent part of speech and making useful data. Some important preprocessing like noise reduction, pre-emphasis, framing is necessary to perform feature extraction. So, this paper performs several preprocessing processes for extracting features from continuous speech data. The preprocessing steps including Noise Reduction, Pre-emphasis, Framing, Window (Most popular Hamming Window), and Normalization which provides the final data for performing feature extraction.

II. Literature Review

A lot of research has done for the process of preprocessing and feature extraction. Most of them perform noise reduction, feature extraction using a single algorithm, which provides less accurate features for specific speech data. But for the overall comparison of accuracy and correctness of processed data depends on the comparative study of available algorithms and the algorithm they used. Nema [4] perform a method of preprocessing where they try to recognize emotion using statistical indicators like The Standard Deviation, The Mean from the musical signal and the normal voice signal. Choi [5] propose a music tagging system for music research comparing experiment of scaling, logarithmic magnitude compression, time-frequency representations and frequency of a music data while Kolokolov [6] perform a preprocessing filtering Logarithmic Spectrum of voice data for Speech Recognition. Kusumoto [7] performs speech preprocessing for a hall or classroom from microphone to loudspeaker by filtering modulation frequency domain and from noisy speech data, Siohan [8] shows that attained Linear Discriminant Analysis (LDA) space is much better than MFCC algorithm. Taal [9] design an algorithm for the near-end listener which improve the intelligibility of listening over frequency and time based on the spectro-temporal auditory model when Yang [10] perform preprocessing for speech synthesis using moving windows, blending the voice for setting amplitude threshold method using polynomial smoothing method and Pitch (fundamental frequency of speech signal) period extraction using MACF and W-AMDF. A pitch extraction method also proposed by Nazrul [11] which reduce non-pitch peaks from raw signal adding rectified DH and also said that their method is better than the NCCF and WAC method. For removing noise, Sambur [12] developed a method called Wiener filter for performing preprocessing which is effective and cheaper, and use to enhance the Linear Prediction Analysis and said that the performance is quite better than others are. Calculating coherence function with Bark sub-band for noise-robust feature extraction, which is a two-channel approach designed by Peters [13] for preprocessing of the nonstationary signal. Bach-end classification technique and front-end feature extraction are studied by Chengalvarayan [14] and proposed that they can reduce 8% more error rate than the traditional model using MFCCs. Hurtado [15] proposed a method for defining an acoustic feature with a noise reduction as side work using Spectral Subtraction algorithm and performing two-channel approach, which provide better result. Three approaches of noise reduction named JAH-RASTA processing, Cepstral Coefficients, and RASTA processing are experimented by Kasper [16] which calculate the first portion of the voice signal and then mapping coefficient and get the best performance by JAH-RASTA processing. Kumar [17] used open-source tools like sox and audacity to remove the noise from speech signals, Karray [18] perform nose cancellation calculation mean noise spectrum from unvoiced speech and subtracted the result from the overall signal spectrum.

Most of the above research is done by a single or multiple algorithm for extracting a specific feature in different data. They do not represent a series of preprocessed method which provide the final and most suitable clean data for any type of further process like this paper. The process is presented in this paper can extract any type of damaged or noisy data only because of the series of process is presented. The main purpose in this contribution is to try this process for Bangla speech which is the only contribution for feature extraction in Bangla Speech. Bangla has a total different accent from English. So it is sometimes difficult to work with Bangla Speech.

III. Methodology

In this research, The preprocessing in the step of Noise Reduction, Pre-emphasis, Framing, Window (Most popular Hamming Window), and Normalization.

Fig.1 shows the preprocessing workflow for feature extraction.

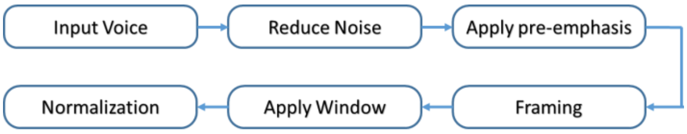

Fig. 1. Preprocessing Workflow.

A. *Noise Reduction*

The first think of speech preprocessing is noise reduction. Because any type of noise in speech is always unnecessary. Algorithm 1 shows the noise reduction process.

Algorithm 1: Noise reduction.

```
Input  :  continuous audio file, A, noise audio file, N
Output :  noise-free audio signal, F
  1    :  Read continuous audio file, A
  2    :  Read noise audio file, N
  3    :  calculates the Fast Fourier Transform(FFT) value of the noise audio clip
  4    :  Based on calculated FFT, some statistics are determined in frequency.
  5    :  based on the statistics, a threshold value is calculated
  6    :  calculating the FFT of the signal audio clip
  7    :  From the comparison of this second FFT and the threshold value, it determines a mask.
  8    :  This is followed by sleeking the mask with frequency and time and apply the mask on the FFT.
  9    :  Output noise-free audio signal, F
```

B. *Pre-Emphasis*

A pre-emphasis filter is very much useful. Because it is used to balance the spectrum of frequency. In some places, the magnitude of higher frequencies is smaller than the magnitude of lower frequencies. This problem is fixed by pre-emphasis. On the other hand, it eliminates numerical problems while doing the operation of the Fourier Transform of a signal and enhances the signal-to-noise-ratio(SNR). The pre-emphasis can be calculated by the equation[19] given below:

$$\mathbf{y[n] = x[n] - \alpha * x[n-1]} \qquad (1)$$

Where (α) is the filter coefficient. Racial values of the coefficient between (0.9) and 1.

C. *Framing*

Before understanding the reason for framing, we need to look at the characteristics of audio signals. The audio signal is a quasi-stationary signal which is one kind of non-stationary signal. That means the statistics of this signal are not static for the whole signal. But an audio signal behaves like a stationary signal for a short interval where the statistics of the signal remains constant. On the other hand we can not apply a Discrete Fourier Transform(DFT) over a non-stationary signal. And DFT is needed to analyze the audio signal which is an important step of our algorithm. To solve this problem we used framing. Here we split the whole audio signal into some frames of frame length 20-40 ms. If we take shorter lengths, we won't have many samples to run the algorithms. The sample rate of our test audio is 16khz. So, with frame length 25ms, we get 0.025*16000 = 400 samples in each frame. But there are some overlaps because while applying the window, we don't want to lose any vital information of the audio signal. So, if the first 400 samples start from 0 index, the next 400 samples will start from 200 index. Here 50% of the signal will overlap. After framing the signal, we apply the window to each frame.

D. Window

Now we apply a window function to each frame. There are various reasons why we need to apply a window function to the frames, and the main reason is notably to counteract the forwardness made by the FFT that the data is limitless and to alleviate spectral leakage. We apply the most popular window called the Hamming window. A Hamming window has been calculated by the equation[20] given below:

$$\mathbf{H(\theta)=0.54+0.46\ cos[(2\pi N)n]} \qquad (2)$$

where $0 \leq n \leq N-1$, n is the window length.

E. Normalization

Now we are moving to the latest step in processing. And It is a technique for modifying the volume of sound to a measured level and to balance the spectrum. On the other hand it improves the Signal-to-Noise (SNR). We apply the most popular normalization called mean normalization. we can be done simply by subtracting the mean of each coefficient from all frames.

IV. Experiment and Output

The experiment is done with three recorded voice data. A portion of Bangla poem which is: "আমাদের ছোট নদী চলে বাকে বাকে, বৈশাখ মাসে তার হাঁটু জল থাকে। পার হয়ে যায় গরু, পার হয় গাড়ি, দুই ধার উঁচু তার ঢালু তার পাড়ি।" recorded in three different place. The recorded samples of data are saved as sample1, sample2, sample3 respectively. Each sample are recorded in three different places. sample1 was recorded in a playground, sample2 was recorded in restaurant and sample3 was recorded at mid-night for noise free environment.

SNR is a ratio that indicates the power of a signal concerning its background noise. Higher SNR means the power of the signal is greater than the power of noise. A comparison of SNR value has been shown in Table I.

TABLE I: SNR value

Filename	Original SNR	Pre-processed SNR
sample1	-2.9178e-05	-3.3478e-08
sample2	-3.8878e-05	-2.7389e-08
sample3	-2.3155e-05	-3.72042e-08

From the table I, it can be seen that there is an increase of SNR for every test data after the preprocessing of raw signals. It has two possible impacts; either the signal is increased or the noise is reduced. As Fig 2, 3 & 4 show that the signal remains the same, it can be considered that the noise has been reduced. The figures show the reduction of noise in a better way. The figure of the raw signal shows that the signal(higher frequencies) is mixed with some noise (lower frequencies). The figure of the signal after noise reduction shows that the noise frequencies are gone. And the higher frequency portions represent the words contained in the whole signal.

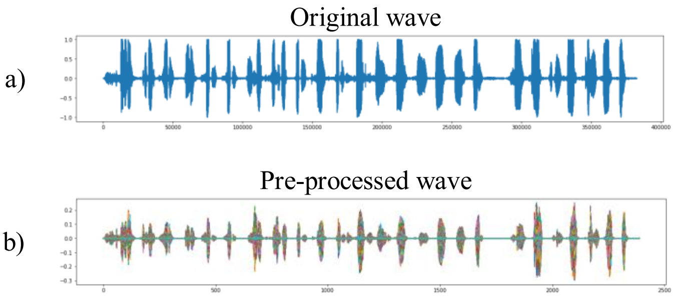

Fig. 2. sample1 speech waveform before and after preprocessing

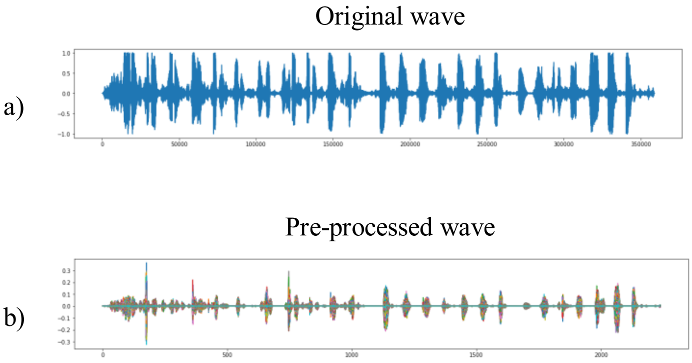

Fig. 3. sample2 speech waveform before and after preprocessing

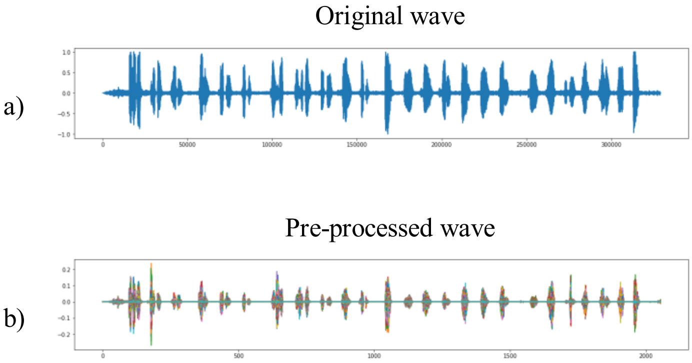

Fig. 4. sample3 speech waveform before and after preprocessing

V. Conclusion

In this paper, we tried to preprocess a continuous Bengali voice signal with a target to make that signal ready for use as an input signal. We have proposed an algorithm that does some preprocessing tasks on the raw signal to make it ready for feature extraction. We tested this algorithm with some raw continuous speech collected in public places. The result of three samples data has been discussed in this paper. In the result section we can see that our proposed method did a good preprocessing. The SNR values of each signal have decreased which indicates that the quality of signals has been improved. Comparing the speech waveform of each wave, given in Fig 2, 3 & 5, we can see that the frequencies of noise have been removed and we are getting a noise-free signal as output. In this paper we have discussed every step with proper instructions and reasons for this method. While performing the above preproceing we face some problem, like it is quite difficult to perform reduction of noise when the recorded data contain more amount of noise than the original speech. Also have some limitation of our contribution, like we can not store and regenerate the original or processed speech data signal after performing framing because framing provide the numpy array value of the processed audio signal.

VI. References

[1] Kuhl, P.K.(2004). Early language acquisition: cracking the speech code. Nat. Rev. Neurosci. 5, 831–843

[2] Yurovsky, Daniel & Yu, Chen & Smith, Linda. (2012). Statistical Speech Segmentation and Word Learning in Parallel: Scaffolding from Child-Directed Speech. Frontiers in psychology. 3. 374. 10.3389/fpsyg.2012.00374.

[3] R. Chengalvarayon, "Time-varying discriminative feature extraction effective for phonetic discrimination," Proceedings of ICICS, 1997 International Conference on Information, Communications and Signal Processing. Theme: Trends in Information Systems Engineering and Wireless Multimedia Communications (Cat., Singapore, 1997) pp. 767-771 vol.2.

[4] Nema, Bashar & Abdul-Kareem, Ahmed. (2017). Preprocessing Signal for Speech Emotion Recognition. Al-Mustansiriyah Journal of Science. 28. 10.23851/mjs.v28i3.48.

[5] K. Choi, G. Fazekas, M. Sandler and K. Cho, "A Comparison of Audio Signal Preprocessing Methods for Deep Neural Networks on Music Tagging," 2018 26th European Signal Processing Conference (EUSIPCO), Rome, 2018, pp. 1870-1874.

[6] Kolokolov, A.. (2002). Signal Preprocessing for Speech Recognition. Automation and Remote Control. 63. 494-501. 10.1023/A: 1014714820229.

[7] A. Kusumoto, T. Arai, T. Kitamura, M. Takahashi and Y. Murahara, "Modulation enhancement of speech as a preprocessing for reverberant chambers with the hearing-impaired," 2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.00CH37100), Istanbul, Turkey, 2000, pp. II853-II856 vol.2.

[8] O. Siohan, "On the robustness of linear discriminant analysis as a preprocessing step for noisy speech recognition," 1995 International Conference on Acoustics, Speech, and Signal Processing, Detroit, MI, USA, 1995, pp. 125-128 vol.1.

[9] C. H. Taal, R. C. Hendriks and R. Heusdens, "A speech preprocessing strategy for intelligibility improvement in noise based on a perceptual distortion measure," 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Kyoto, 2012, pp. 4061-4064.

[10] Yang Fan, Xu Sun-hua, Liu Ming-hui and Pan Guo-feng, "Research on a new method of preprocessing and speech synthesis pitch detection," 2010 International Conference On Computer Design and Applications, Qinhuangdao, 2010, pp. V1-399-V1-401.

[11] N. I. Nazrul, M. T. H. Setu, S. Hussain and K. Hasan, "An effective speech preprocessing technique for normalized cross-correlation pitch extractor," Proceedings of the 3rd IEEE International Symposium on Signal Processing and Information Technology (IEEE Cat. No.03EX795), Darmstadt, Germany, 2003, pp. 749-752.

[12] M. Sambur, "A preprocessing filter for enhancing LPC analysis/Synthesis of noisy speech," ICASSP '79. IEEE International Conference on Acoustics, Speech, and Signal Processing, Washington, DC, USA, 1979, pp. 971-974.

[13] M. Peters, "Binaural Bark subband preprocessing of nonstationary signals for noise robust speech feature extraction," Proceedings of the IEEE-SP International Symposium on Time-Frequency and Time-Scale Analysis (Cat. No.98TH8380), Pittsburgh, PA, USA, 1998, pp. 609-612.

[14] R. Chengalvarayan, "Linear trajectory models incorporating preprocessing parameters for speech recognition," in IEEE Signal Processing Letters, vol. 5, no. 3, pp. 66-68, March 1998.

[15] J. E. Hurtado, G. Castellanos and J. F. Suarez, "Effective extraction of acoustic features after noise reduction for speech classification," Modern Problems of Radio Engineering, Telecommunications and Computer Science (IEEE Cat. No.02EX542), Lviv-Slavsko, Ukraine, 2002, pp. 245-248.

[16] K. Kasper, H. Reininger and D. Wolf, "Exploiting the potential of auditory preprocessing for robust speech recognition by locally recurrent neural networks," 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing, Munich, 1997, pp. 1223-1226 vol.2.

[17] A. Kumar, H. Hemani, N. Sakthivel and S. Chaturvedi, "Effective preprocessing of speech and acoustic features extraction for spoken language identification," 2015 International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM), Chennai, 2015, pp. 81-88.

[18] L. Karray and L. Mauuary, "Improving speech detection robustness for wireless speech recognition," 1997 IEEE Workshop on Automatic Speech Recognition and Understanding Proceedings, Santa Barbara, CA, USA, 1997, pp. 428-435.

[19] Speech processing pre-emphasis: how does it work? - Mathematics Stack Exchange. Retrieved April 19, 2020, from https://math.stackexchange.com/questions/44216/speech-processing-pre-emphasis-how-does-it-work

[20] Hamming Window - ScienceDirect Topics. Retrieved April 19, 2020, from https://www.sciencedirect.com/topics/engineering/hamming-window