Journal of Engineering Sciences
Vol 11, Issue 6,June/2020
ISSN NO:0377-9254

Speech Easy using Machine Learning and Virtual Reality

Ch. Chandra lekha[1], G. Swathi[2] and G Niharika[3]

UG Scholars, Department of CSE
Divya Kumari Tankala, Assistant Professor, Department of CSE, G.Narayanamma Institute of Technology and Science (for women), Telangana, India.
Email-id: divya.aug9@gmail.com

***Abstract*— Stuttering has no Effective Cure: Even after it affecting 96 Million people of all ages all over the world, stuttering still doesn't have any effective cure yet. Stammering is a speech disorder in which the flow of speech is disrupted by involuntary repetitions and prolongations of sounds, syllables, words or phrases as well as involuntary silent pauses or blocks in which the person who stutters is unable to produce sounds. Stuttering is more than just a speech disorder; it has become a way of life which is a necessity to be changed. We used "Mirror Neurons "to give video and manipulated audio to stimulate these neurons feedback using ML trained models.**

***Keywords*— Virtual Reality, Machine learning, Stuttered Speech recognition**

1. INTRODUCTION

In today's world there are millions of persons suffering from various speech disorders like stuttering, lisp, etc., this often renders them unable to utilize certain things that we take for granted, like speech recognition systems. Stuttering is one such speech disorder affecting the fluency of speech. It begins during childhood and, in some cases, lasts throughout life. The disorder is characterized by disruptions in the production of speech sounds, also called "disfluencies." Most people produce brief disfluencies from time to time. For instance, some words are repeated and others are preceded by "um" or "uh." Disfluencies are not necessarily a problem; however, they can impede communication when a person produces too many of them. In most cases, stuttering has an impact on at least some daily activities. The specific activities that a person finds challenging to perform vary across individuals. For some people, communication difficulties only happen during specific activities, for example, talking on the phone or talking before large groups, utilizing everyday tools that use speech as inputs.

Currently, the speech recognition systems have a great accuracy for fluent speech but are unable to recognize speech with repetitions or long involuntary pauses, i.e. stuttering. This is mainly because the systems are created to stop the identification process when a pause is encountered. Also, these systems are trained with proper words without any repetitions and so, when it encounters a stuttered speech, it is unable to identify the words, since it hasn't been trained to do so. Many people have detected stuttering from speech samples; however, they haven't corrected the sample. They have use ANN, HMM, SVM to name a few and advance DSP to remove noise from the samples and correct them. Our project aims to detect as well as correct these stuttered speech samples on the fly and then give the corrected speech sample devoid of stuttering. We will be using neural networks and DSP to detect and correct the speech. This system can then be integrated with phones and laptops and help people suffering from this
Speech impairment to control their devices with speech, the same way as most of the population in today's world does. Thus, our main aim is to help these people by making certain already accessible tools available to them, without worrying about their speech impairment.

2. Approach of Speech Easy

The common way to recognize speech is the following: we take waveform, split it on utterances by silences then try to recognize what's being said in each utterance. To do that we want to take all possible combinations of words and try to match them with the audio. We choose the best matching combination.

First of all, it's a concept of features. Since number of parameters is large, we are trying to optimize it. Numbers that are calculated from speech usually by dividing speech on frames. Then for each frame of length typically 10 milliseconds we extract 39 numbers that represent the speech.

That's called feature vector. They way to generates numbers is a subject of active investigation, but in simple case it's a derivative from spectrum. Second, it's a concept of the model. Model describes some mathematical object that gathers common attributes of the spoken word. In practice, for audio model of Sen one is Gaussian mixture of its three states - to put it simple, it's a most probable feature vector. From concept of the model the following issues raised - how good does model fits practice, can model be made better of its internal model problems, how adaptive model is to the changed conditions.

The model of speech is called Hidden Markov Model or HMM, it's a generic model that describes black-box communication channel. In this model process is described as a sequence of states which change each other with certain probability. This model is intended to describe any sequential process like speech. It has been proven to be really practical for speech decoding. Third, it's a matching process itself. Since it would take a huge time more than universe existed to compare all feature vectors with all models, the search is often optimized by many tricks. At any points we maintain best matching variants and extend them as time goes producing best matching variants for the next frame.

The main aim of our system is to remove and correct stuttered speech, however, in order to better learn how speech recognition works, we decided to implement a small speech recognition system of 10 words on our own. Our small program learns 10 words and recognizes them when spoken to a great degree of accuracy. It is based on Linear Predictive Coding [LPC] and a simple implementation of gradient decent and linear regression, both are integral part of neural networks. We have trained the neural network to recognize 10 simple words, they are “One, Two, Three, Four, Five, Six, Seven, Eight, Nine and Zero”. The training set contains 120 samples; each word has been spoken 12 times. The samples were created by both of us and include various ways in which the word is spoken.

The system works well for a small learning data set. However, in order to make our main project robust, we will be using Googles Speech API in detection of the corrected stuttered speech because it is not feasible to create the whole speech recognition. Thus, Googles API will be used to make the program robust.

3. Methodology

Usually, the classification is done by processing the entire audio file manually. This manual method would lead to more pressure on humans as the number of files increases. To solve the problem the system is divided into multiple sections consisting of the various possible methods.

As shown in Figure 1, Speech is taken as input and is pre-processed, and noise is removed and is then processed using ml algorithms where detection is one of the main steps in the recognition of any speech sample. There are various methods to detect stuttered speech like Artificial Neural Networks, Hidden Markov Models, Standard Vector Machines and Digital Signal Processing.

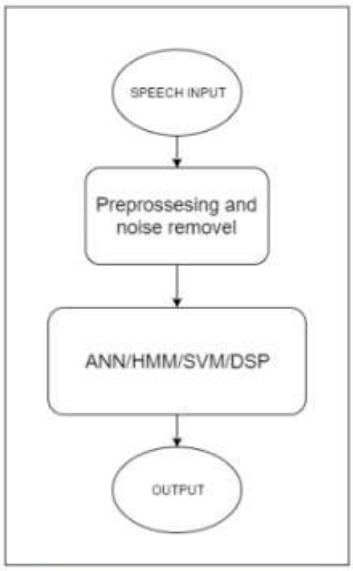

Figure 1: Architecture of Speech Easy

Initially started off with recording voice samples, next it is converted into .wav file since it is a format which is compatible with both, MATLAB and Python. After that, put the file in its matrix format in MATLAB, the filtering of stuttering could be started.

Now, the speech is divided into small frames. Each frame is checked for the maximum amplitude that it obtains, and if the value is greater than the threshold, the matrix values for that frame are copied to a new variable. This is done for the entire speech (or all samples) and after the process is complete, the new variable has a speech sample which is mostly free from stuttering. Back Propagation algorithm applied to have maximum

amplitude as its input and the threshold value as its output.

The algorithms for the correction and recognition of stuttered speech are as follows:

- Take a speech sample from user (5 seconds with 8000Hz sampling frequency).
- Insert a speech as a matrix in a variable (file).
- Obtain the maximum amplitude of the speech.
- Pass the maximum amplitude to the python script to compute a threshold value using neural networks
- Divide the speech samples into short frames of equal length.
- Analyses each frame and if the max value of the frame is greater than the threshold value copy the frame onto a new signal variable.
- Once all frames have been analyzed, convert the variable into a .wav file.
- Pass the .wav file to another python script, which will recognize the clean audio sample using Google's speech recognition API.
- Build a neural network specifying number of inputs, hidden layers and SupervisedDataSet function.
- Now, use the activate function and pass the current input amplitude to it, to get the threshold value

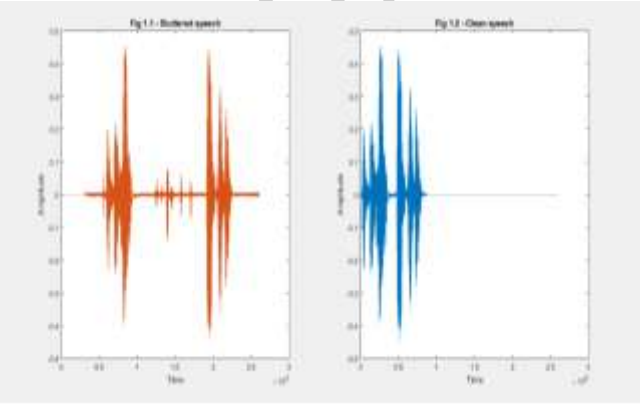

Figure 2: Stuttered speech and clean speech

Figure 3: Interface for to practice stammering videos.

4. Conclusion

The stuttered speech correcting is able to achieve an accuracy of 84% on 50 test samples. Our approach further improved by increasing the number of training samples. Our approach helps in getting favorable outcomes. We could also use another parameter along with amplitude to better detect and correct the stuttered speech. Stuttering is only one of the common speech disorders, we could also implement the same with other speech impediments like lisp, etc.

References

[1] Cohen, J., Cohen P., West, S.G., & Aiken, L.S. (2003). Applied multiple regression/correlation analysis for the behavioral sciences. (2nd ed.) Hillsdale, NJ: Lawrence Erlbaum Associates

[2] Draper, N.R.; Smith, H. (1998). Applied Regression Analysis (3rd ed.). John Wiley. ISBN 0-471-17082-8.

[3] K. M. Ravikumar, Balakrishna Reddy, R. Rajagopal, and H. C. Nagaraj, "Automatic Detection of Syllable Repetition in Read Speech for Objective Assessment of Stuttered Disfluencies", International Journal of Electrical and Computer Engineering Vol:2, No:10, 2008

[4] Chee, Lim Sin, Ooi Chia Ai, Sazali Yaacob and Jejawi Perlis. "Overview of Automatic Stuttering Recognition System." (2009).