Stuttered Speech Recognition

Prof. Sankar Ganesh, Anshuman S. Dhamoon, Sarthak Khanna

VIT University

Vellore, Tamil Nadu

Abstract - The aim of this project is to come up with a new algorithm to enhance speech recognition for people suffering from stuttering. The basic idea is to first remove stuttering from the sample by using the amplitude threshold obtained from neural networks and then passing the clean sample through Google's Speech Recognition API. The major problem until now is that a system stops recognizing after a pause is encountered in the speech and hence, the average accuracy of stuttered speech recognition is around 70%. With a new algorithm which will take into account the words or characters after the pause and then use that also for recognition, the accuracy can be improved.

Introduction

Stuttering is one of the major speech impediments that people are suffering from, in today's world. It is a disorder which affects the fluency of speech by involuntary repetitions or prolongations of words, syllables, etc or involuntary silent intervals. Of all these, repetition is the most common and prominent characteristic of stuttering. In today's world, about 1% of the total population (or 70 million people) stutter, which shows that it is a major problem.

A few institutions have researched on detection of stuttering using acoustic level analysis methods. A better method than the above is to create a stuttering detection system based on a large vocabulary of continuous speech recognition, but a technique like this is very resource and power intensive, and hence, cannot be used practically.

We observed, that when a person stutters, there is a decrease in the amplitude in the person's voice signal and we used this, to eliminate the repetitions, elongations and silent intervals to produce a better speech recognition system.

Method

A. *Previous Methods*

Most of the work done in stutter detection uses MFCC and fuzzy logic to extract features from the given speech samples and use fuzzy logic, along with neural networks to generalize the solution. Some have used HMM for stuttered speech recognition, by dividing the speech samples into phonemes and assigning probabilities for the different decisions the system is supposed to take, in case a pause is encountered, a phoneme is repeated or it is elongated.

A major drawback of all these approaches is the amount of computation power that will be required for the above techniques. When using just neural networks for stuttered speech recognition, it is impossible to come up with an efficient solution since there is no limited number of sentences than can be put into the system with all its permutations and combinations, to detect a sample every time.

Similarly, when using HMM with MFCC, a lot of different observations will have to be predicted and the greater the dataset, more will be the number of observations for HMM to predict from and it might become less accurate.

B. *Proposed Solution*

A simple solution to overcome the enormous computation power and gigantic datasets, is to use amplitude as a factor, to remove stuttering. We observed that when a person stutters, the amplitude of his speech drops significantly, when compared to the rest, and hence, could be used to filter out the stutter from the sample.

So, we checked a lot of stuttered speech datasets for the above, and noticed that the threshold needed for removing the stutter from the sample largely depended on the maximum amplitude observed in the sample.

We created a neural network that would only take the maximum amplitude of the sample as an input and give the required threshold as an output. We initially selected the thresholds manually, by looking at the plots of the speech samples and after acquiring a large enough dataset, used those values to train a neural network using back propagation algorithm. After the training was complete, the threshold output given by the network was accurate enough for us to clean the stuttered sample enough, that it could be easily recognized.

Since we eliminated the stuttering, we now simply had to pass a normal speech to an already existing system. Google is the

leading company when it comes to speech recognition and they have spent years, perfecting it for different types of words, sequences, accents, etc. And the system that they have created has been open sourced for the use of general public.

So, after acquiring the clean speech, we created a python script to which we could pass this speech, from MATLAB. Then we call the services of Google's system, pass it our speech and the system does the rest of the work by analyzing the speech and returning the result.

Algorithm

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

Stuttered Speech Recognition

1. Take a speech sample from user.
2. Insert a speech as a matrix in a variable (file).
3. Obtain the maximum amplitude of the speech.
4. Pass the maximum amplitude to the python script to compute a threshold value using neural networks*.
5. Divide the speech samples into short frames of equal length.
6. Analyze each frame and if the max value of the frame is greater than the threshold value, copy the frame onto a new signal variable.
7. Once all frames have been analyzed, convert the variable into a .wav file.
8. Pass the .wav file to another python script, which will recognize the clean audio sample using Google's speech recognition API.

***Neural Networks**

1. Create a python script and import pybrain module to use the functions buildNetwork, SupervisedDataSet and BackpropTrainer.
2. Define a function which will take the maximum amplitude of the speech signal as an input.
3. Build a neural network specifying number of inputs, hidden layers and outputs, using buildNetwork function.
4. Define the type of dataset, again specifying the number of inputs and outputs using SupervisedDataSet function.
5. Add the training set to supervised dataset (at least 50), which comprises of maximum amplitude as input and

the threshold for that amplitude as the output.

6. Call the BackpropTrainer function and pass the network and dataset to specify the algorithm being used to train the network (Back propogation)
7. Call the trainUntilConvergence function to train the data till it converges to a particular set of weights.

Now, use the activate function and pass the current input amplitude to it, to get the threshold value and return the given value to the MATLAB file.

Simulated Results

A. Plots

The following plots show the stuttered speech and speech after the stutter has been removed for three different samples.

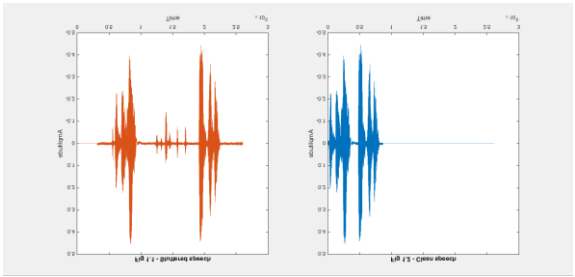

Figure 1: Stuttered and Clean Sample A

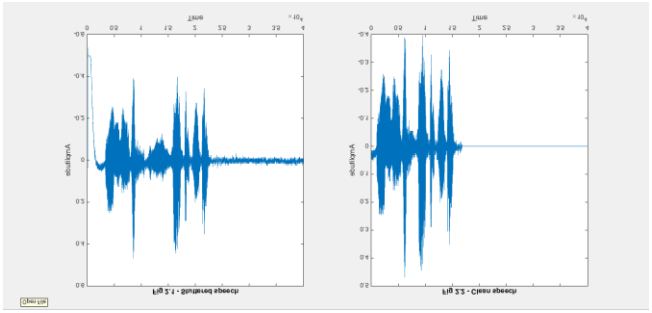

Figure 2: Stuttered and Clean Sample B

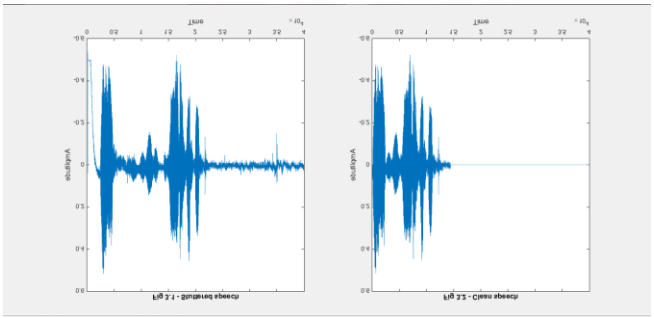

Figure 3: Stuttered and Clean Sample C

From the above plots, we can observe the difference in the amplitudes when stuttering occurs and how we used threshold values, which we obtained from the neural network that we trained, are used to clean the sample.

B. Tabulated Result

We tested using a total of 50 test cases, with 42 of them being detected correctly. A summary of the result obtained is shown in the table below.

Total # of test samples	Predicted correctly	Predicted incorrectly
50	42	8

Table 1: Results obtained

$$K = P/n$$

Where,

***K* -** Accuracy

***P* -** Correctly predicted test samples

***n* -** Total samples

Therefore, putting values form Table 1 into the equation we get an accuracy of:

Accuracy = 84.0 %

Conclusion

The increasing usage of speech recognition systems by people has led to the ease of access in their day to day lives. People use personal assistants like Apple's Siri, Microsoft Cortana or Google Now and make their lives easier, however people with speech impairments like stuttering cannot benefit from these services. In order to make these above mentioned applications more universal we worked a project that would actually help in solving a real world problem and we were successful in getting favorable outcomes. Not a lot of work has been done using amplitude as the primary technique of elimination of stutter, and this makes our project novel and research oriented.

Our stuttered speech correcting program was able to achieve an accuracy of 84% on 50 test samples. This shows promise for our approach and the accuracy can further be improved by increasing the number of training samples that we used in the training set.

Future Scope

1. Integration of our solution with already existing speech recognition services across all platforms like PC, mobile, etc. Which would enable the affected persons to use speech recognition tools and services even with their stutter.
2. We can further increase the accuracy and effectiveness of our technique by acquiring more data samples from affected individuals, which would result in a larger training set, thus making our neural network more robust.
3. We could also use another parameter along with amplitude to better detect and correct the stuttered speech.

References

1.Overview of Automatic Stuttering Recognition System, Lim Sin Chee, Ooi Chia Ai, Sazali Yaacob, Universiti Malaysia PerlisJalan Kangar-Arau 02600 Jejawi Perlis

2. Robert M. Gray, IEEE Signal Processing Society, Distinguished Lecturer Program

3. Structure Analysis of Speech Signals and Filteringthem from Noise, Anastasiya Andreevna Kuznetsova, Admiral Makarov State University of Maritime and Inland Shipping, Saint-Petersburg, Russia

4.Automatic Detection of Syllable Repetition in Read Speech for Objective Assessment of Stuttered Disfluencies, K. M. Ravikumar, Balakrishna Reddy, R. Rajagopal, and H. C. Nagaraj

5. Stuttered Speech Recognition for Robotic Control - G. Manjula, Dr. M. Shiva Kumar

6.http://www.asha.org/public/speech/disorders/stuttering/

7.https://en.wikipedia.org/wiki/Stuttering