International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878, Volume-8 Issue-5, January 2020

Automatic Speech Recognition with Stuttering Speech Removal using Long Short-Term Memory (LSTM)

S.Girirajan, R.Sangeetha, T.Preethi, A.Chinnappa

***Abstract*: Stuttering or Stammering is a speech defect within which sounds, syllables, or words are rehashed or delayed, disrupting the traditional flow of speech. Stuttering can make it hard to speak with other individuals, which regularly have an effect on an individual's quality of life. Automatic Speech Recognition (ASR) system is a technology that converts audio speech signal into corresponding text. Presently ASR systems play a major role in controlling or providing inputs to the various applications. Such an ASR system and Machine Translation Application suffers a lot due to stuttering (speech dysfluency). Dysfluencies will affect the phrase consciousness accuracy of an ASR, with the aid of increasing word addition, substitution and dismissal rates. In this work we focused on detecting and removing the prolongation, silent pauses and repetition to generate proper text sequence for the given stuttered speech signal. The stuttered speech recognition consists of two stages namely classification using LSTM and testing in ASR. The major phases of classification system are Re-sampling, Segmentation, Pre-Emphasis, Epoch Extraction and Classification. The current work is carried out in UCLASS Stuttering dataset using MATLAB with 4% to 6% increase in accuracy when compare with ANN and SVM*.

***Keywords* : ASR, Dysfluency, repetition, prolongation, LSTM, MFCC Feature Extraction.**

I. INTRODUCTION

Human uses Speech to communicate, express his thoughts and feelings to other human beings. Nowadays human controls the electronic devices with help of speech through the technology called Speech recognition. Speech recognition is an easiest, most natural way of human communication to give inputs to the devices or to control it. In recent years speech recognition popular way of controlling variety of application including medical application, industrial robotics, home automation, defence, machine translation etc. Initially speech recognition system used to recognize single word or number for which you have to maintain pause between each word and number [1, 2]. Since lot of research carried out in speech recognition right now we can able to recognize continuous speech like conversationally paced speech.

Speech recognition systems digitize, separate speech from background noise, finds the phoneme from the audio frames, compare the phoneme to predict the word and finally based on the language properties next word will be predicted in a speech sequence [3, 4, 5 and 6].

Stuttering is a speech dysfluency disorder, such as sound/syllable reiteration or prolongations in the expression of short speech components and words [7]. Speaker comprehends what to state however is unable to state it due to an automatic tedious prolongation or end of a sound. The speech dysfluency affects the ASR in several ways. Dysfluency like repetition generates longer utterance without any proper meaning. Usually well formatted dataset are used to train the ASR system but the dysfluency produces irrelevant content which will mismatch the training and testing data that leads to poor transcription [8]. For example consider “I we-we-went to shopping uhmmm yesterday”in such a situation it is more complex for ASR to understand the sounds like “uhmmm” so needs to make some word addition, cancellation and dismissal to increase the quality of transcription.

In the proposed system first we extract the human voice separately from the noise in the given audio speech signal with the help of Mel-frequency cepstral coefficients (MFCCs) feature extraction, then classify the stuttering and non-stuttering speech signal separately by using RNN.

After classifying stuttering and non-stuttering speech signal we remove the stuttering speech signal then the audio signal is passed to the Google cloud speech to text model to test the word error rate.

II. RELATED WORK

Automatic Stuttering speech recognition is usually carried using some of the classification techniques like ANN, HMM, SVM.

Manuscript published on January 30, 2020.
* Correspondence Author
S.Girirajan*, Department of Computer Science and Engineering, SRM Institute of Science and Technologyy, Kattankulathur, India. Email: girirajans.cse@gmail.com
R.Sangeetha, Department of Computer Science and Engineering, SRM Institute of Science and Technologyy, Kattankulathur, India. Email: sangeethacse1991@gmail.com
T.Preethi, Department of Computer Science and Engineering, SRM Institute of Science and Technologyy, Kattankulathur, India. Email: preethiits89@gmail.com
A.Chinnappa, Department of Computer Science and Engineering, SRM Institute of Science and Technologyy, Kattankulathur, India. Email: chinnacse@gmail.com

This segment introduces a review of past works found in the literature which focuses on how the Automatic Stuttering speech recognition is being carried out.

A. HIDDEN MARKOV MODELS (HMMs)

Author (s) can send paper in the given email address of the journal. There are two email address. It is compulsory to send paper in both email address.

HMM is one of the most commonly used techniques in ASR, especially in Automatic stuttering speech recognition like prolongation, repetition of words. HMM is a technique used to predict the phones from the given set of speech signal. The properties of HMM are well understood, with many sophisticated and efficient algorithms for training and decoding developed around it. These factors have made HMM incredibly popular in ASR, and have resulted in huge improvement over Dynamic Time Warping (DTW) [9 10].

Recognise the utterance "eks" ("X") using HMM is shown in Figure 1. The audio signals are divided into frames or segment during pre-processing. Each segment is considered as a states of the HMM which is denoted as circles. The arrow in states denotes the transitions between each state.

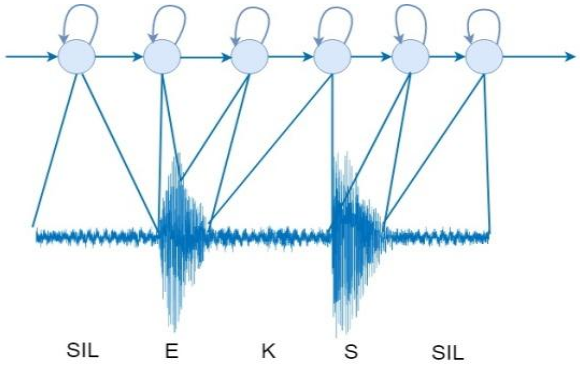

Fig. 1.Fig. 1 Recognise the utterance "eks" ("X") using HMM

Tian-Swee et al proposed an stuttering recognition system for the children's with the help of HMM. HMM model is trained with voice pattern of stutter and non-stutter children's speech to classify the stutter speech. The process achieved 96% accuracy in finding the non-stutter speech and 90% accuracy in stutter speech recognition [11].

B. SUPPORT VECTOR MACHINE (SVM)

SVM separates given set of data points of two types into two separate groups using hyperplane. Data points that are nearest to the hyperplane is known as support vectors [12 13].

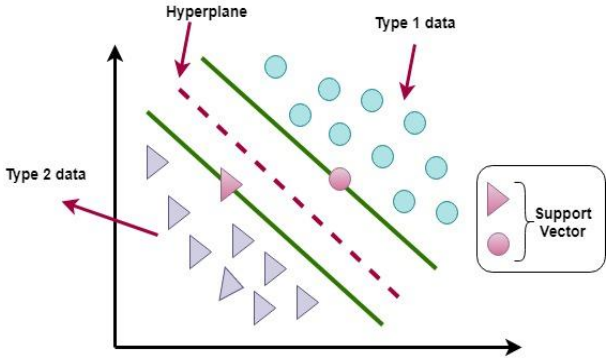

Fig. 2.Fig 2. Support Vector Machine

Ravikumar et al [14] used SVM for implementing automatic stutter speech recognition. The stutter speech samples of fifteen adults were collected within which 12 samples are utilized for training and 3 samples are utilized for testing purpose. This work performs higher with the 94.35% accuracy. That is higher in comparison with their previous work.

C. ARTIFICIAL NEURAL NETWORKS (ANNs)

ANN is one of the major tool used in machine learning for finding patterns which are too complex for humans. As the "neural" part of their name suggests, they are brain-inspired systems which are intended to replicate the way that we humans learn. Neural networks consist of input and output layers, as well as a hidden layer consisting of units that transform the input into the output.

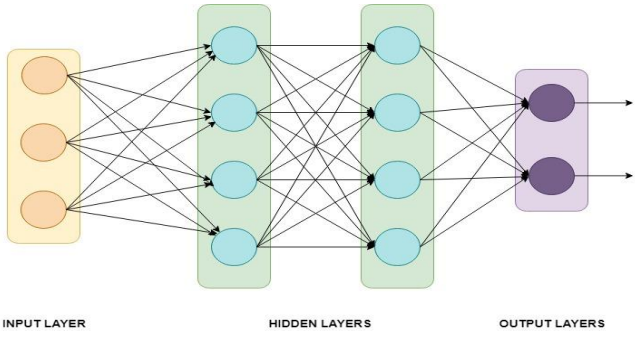

Fig. 3.Fig. 3. Artificial Neural Networks

Technique to detect the dysfluency in speech signal and to classify stutter and non-stutter speech ANN are widely used. ANN was trained with different features like MFCC, zero crossing rates (ZCR), Pitch etc. Such a combination gave better accuracy of 94.52% for repetition and 96.71 % for prolongation.

When compare with other combination of features MFCC along with ANN shows an 88.29% average accuracy [15].

III. THE SPEECH CORPUS

Stuttering speech recording is released by University College London's Archive in three versions UCLASS Releases One, Two and UCLASS-FSF on 2004 and 2008 respectively. The recording is collected between the age groups 18 to 45 with equally divided amoung male and female. The UCLASS one and two recordings were made in normal speaking conditions and the UCLASS-FSF was made when the sound of the speaker's voice was altered as he or she spoke. The data and software are freely available to anyone for research and teaching purposes. The dataset were available in both MP3 as well as WAV format along with the Orthographic and Phonetic transcription [16].

IV. METHODOLOGY

In the proposed system first we classify the stutter and non-stutter speech separately then speech signal is corrected and stored as shown in Fig 1. Secondly corrected speech is used to identify stutter speech in real world ASR.

The procedure of stammered speech recognition is isolated into six phases: Segmentation, Feature Extraction, Classification, predicting correct word match, correcting the audio single, stored in speech database.

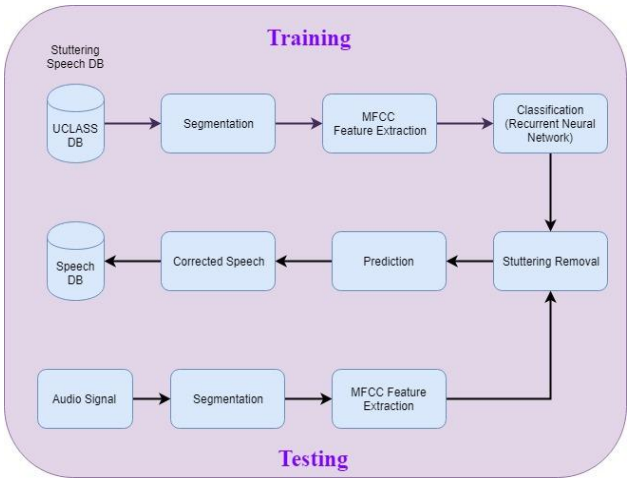

Fig. 4.Fig. 4. Automatic Stuttering Speech Recognition

A. SEGMENTATION

Segmentation of speech signals is widely used for speech analysis and recognition purposes. Segmentation is a process in which continuous speech is divided into smaller units having boundaries with fine resolutions. The smaller units can be of any forms like word, phone and syllable. Segmentation can be carried out in both manually as well as automatically. Automatic segmentation is better when compare with Manual segmentation. Various ways to perform automatic segmentations are Fourier Transform, Short Term Energy, Minimum Phase Group Delay Method, Wavelet Method, Discrete Wavelet Transform (DWT) and Word Chopper Technique [16]. In this proposed system we used DWT method for segmentation since it uses frequency and time concurrently due to that computing the threshold value will be accurate [17]. The DWT is defined as

$$W(i,j) = \Sigma\Sigma x(j) 2^{-i/2} \Psi(2^{-i}k - j) \quad (1)$$

i,j,k are integer values.

$$\Psi_{\tau,\alpha}(t) = \alpha^{-\frac{1}{2}} \Psi\left(t - \frac{\tau}{\alpha}\right) \quad (2)$$

Where $\Psi(t)$ represents mother wavelet or the prototype filter.

τ - Interpretation variable.

α - scaling variable.

$\alpha^{-1/2}$ - standardization term.

B. FEATURE EXTRACTION

Feature extraction is used to convert the acoustic signal into a sequence of acoustic feature vectors that carry a good representation of input speech signal. Researchers working on this area discovered that if the speech signal is observed using a very small duration window, the speech content in that small duration appears to be more or less stationary. That brought in the concept of short-time processing of speech. In this technique, a small duration window is considered for processing at a time. This small duration is called a frame. To process the whole speech segment, we need to move the window from beginning to end of the segment consistently with equal steps, called shift. Based on the frame-size and frame-shift we can calculate M frames. For each of the frames, MFCC coefficients are computed. Steps in calculating MFCC are given below.

- Take the Fourier transform of (a windowed excerpt of) a signal.
- Map the powers of the spectrum obtained above onto the mel scale, using triangular overlapping windows.
- Take the logs of the powers at each of the mel frequencies.
- Take the discrete cosine transform of the list of mel log powers, as if it were a signal.
- The MFCCs are the amplitudes of the resulting spectrum.

C. Pre-Emphasis

The Strategic separation is sustained from numerical issues throughout the Fourier transform, Signal-to-Noise Ratio (SNR) raise by stabilizing the frequency spectrum. In the equation (3) first order filter is used to apply pre-emphasis filter on signal x

$$y(t) = x(t) - \alpha x(t-1) \quad (3)$$

D. Framing

Speech signals are sliced into frames after pre-emphasis based on stability of the signal. The method of reasoning behind this progression is that frequencies in a signal change after some time, so much of the time it doesn't be clear to do the Fourier transform. To overcome these issues adjacent frames are joined together to form an excellent frequency outline after Fourier transformation.

E. Window

Windowing isolates a single interval of the signal for processing. To reduce the noise present at the starting and ending of the frames, window functions are used. Finite set of data is captured in the time domain; there is spectral bandwidth due to this truncation. This is a rectangular window. Using equation (4) hamming window function is applied to each frame.

$$w[k] = 0.53836 - 0.46164\cos\left(\frac{2\pi k}{M-1}\right) \quad (4)$$

where, $0 \le k \le M-1$, M is the sample in each frame.

F. Fourier-Transform

Fourier transform treats signals as periodic in nature. We apply FFT (Fast Fourier Transform) on to a signal that will assumes the chunk of signal and then repeat itself infinitely to compute the power band of each frame. Equation (5) used to calculate the power spectrum.

$$P = \frac{|FFT(x_i)|^2}{M} \quad (5)$$

Where, x_i is the ith frame of signal x.

G. Long Short-Term Memory (LSTM) Classifier

LSTM is basically considered to avoid the problem of vanishing gradient in RNN. Previously computed inputs are available in LSTM memory cell until forgot gate is open and input gate is closed. Without changing the cell content output layer can able to switch on or off due to better control provided by output gate.

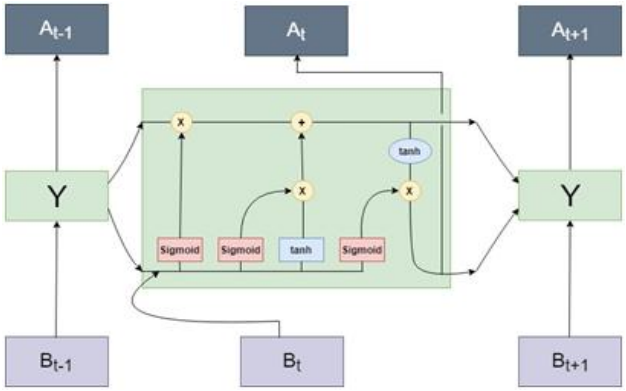

Fig. 5.Bt-input At-output Y-Neural Network

Fig. 6. Fig. 5. Long short-term Memory (LSTM)

Dropout can be applied to ignore previously computed inputs that is not required further in LSTM memory cell. The sigmoid layer takes the input B(t) and A(t-1) and decides which parts from old output should be removed.

The next step is to decide and store information from the new input B(t) in the cell state. From the new input tanh layer creates a vector of all the possible values. The new cell state will be updated after multiplying these two values. The old memory c(t-1) will be combined with the new memory to get c(t).

ALGORITHM

1. Train the LSTM network with stuttering speech dataset.
2. By using simple thresholding technique removes stuttering and absence of speech segments from the given audio files.
3. Extract sequences of functions consisting of Gammatone Cepstral Coefficients (GTCC), pitch, harmonic ratio, and several spectral shape descriptors from the given audio file.
4. LSTM model is trained with sequence of functions.
5. Test and display the classifier's accuracy on the training data.
6. Break the stuttering from these files, create feature sequ ences, transfer them through the network, and check the ir reliability by comparing the speakers ' expected and a ctual speech.

V. EXPERIMENTAL RESULTS

A. Recognition of Stuttering and Non-Stuttering intervals

For evaluating the proposed method, we have used LSTM based stuttering and non-stuttering Classification method. The evaluation was implemented on the UCLASS Stuttering speech database. A total of 10 hrs 30 mins stuttering speech database were splitted into the ratio of 3:1 for training and testing. For Feature extraction 100ms and 200ms were selected as windows size and 0.8, 1 and 1.2 thershold were selected for detecting the stuttering and non-stuttering boundaries. Table 1 represents the Accuracy of the model indicated by precision, recall and F1score. Precision represents the percentage of relevant. Recall refers to the percentage of total relevant results correctly classified. Harmonic mean of precision and recall is denoted by F1 score.

$$precision = \frac{True\ positive}{True\ positive\ +\ False\ positive} \quad (5)$$

$$Recall = \frac{True\ positive}{True\ positive\ +\ False\ Negative} \quad (6)$$

$$Accuracy = \frac{True\ positive + True\ Negative}{Total} \quad (7)$$

$$F1\ score = 2 * \frac{precision * recall}{precision + recall} \quad (8)$$

TABLE 1 A:
STUTTERING AND NON-STUTTERING BOUNDARY DETECTION

Thershold		FD=32 , WS=100 SS=50		FD=384, WS=100 SS=50	
		AM	LM	AM	LM
0.8	Precision	0.27	0.49	0.26	0.50
	Recall	0.34	0.63	0.31	0.60
	F1 Measure	0.30	0.55	0.28	0.54
1	Precision	0.28	0.51	0.26	0.52
	Recall	0.33	0.62	0.30	0.59
	F1 Measure	0.30	0.56	0.28	0.54
1.2	Precision	0.29	0.53	0.26	0.53
	Recall	0.33	061	0.29	0.58
	F1 Measure	0.31	0.57	0.27	0.55

TABLE 1 B:
STUTTERING AND NON-STUTTERING BOUNDARY DETECTION

Thershold		FD=32, WS=200 SS=50		FD=384, WS=200 SS=50	
		AM	LM	AM	LM
0.8	Precision	0.40	0.69	0.35	0.56
	Recall	0.40	0.67	0.39	0.67
	F1 Measure	0.40	0.67	0.37	0.64
1	Precision	0.41	0.70	0.36	0.62
	Recall	0.38	0.63	0.38	0.65
	F1 Measure	0.39	0.66	0.37	0.63
1.2	Precision	0.44	0.72	0.38	0.64
	Recall	0.35	0.57	0.37	0.61
	F1 Measure	0.39	0.64	0.37	0.62

FD Feature Dimension, WS Window Size, SS Shift Size, AM Accurate Match

LM Limited Match

Each individual audio file length will be more than 3 minutes so each audio file is divided into small chucks otherwise called frames. Splitting of audio file is set to constant 5s to avoid saturation in LSTM. We set 2 hidden layers of size 128 and 256 respectively with dropout rate 0.2 and learning rate 0.00006. Then calculated the Mean Absolute Error (MAE) along with correlation coefficient r shown in table 2.

TABLE 2

MAE AND CORRELATION COEFFICIENT

No. of Hidden Layers	Size	MAE/r
Two	128	6.24/0.76
Two	256	6.10/0.77

Repetitions of the words in speech signal are identified by using the decision factor. Decision factor is calculated by multiplying stuttering and non-stuttering intervals with computed Euclidian distance of linear predictive coding as follows

Decision Factor = intervals * Euclidian distance of LPC

After classifying the stuttering and non-stuttering speech separately we removed the stuttering speech signal using decision factor then predicted the correct speech signal for stuttering speech by adding, deleting or substituting a word and stored them in database. In the testing phase we predicted the corresponding text for given stuttering speech signal by using the corrected signal stored in the database. The table 3 shows the comparison of proposed stuttering speech recognition with the SVM and ANN based word error rate.

TABLE 3 WORD ERROR RATE

	SVM	ANN	LSTM
Replacement	29.20%	20.90%	19%
Addition	10.30%	6.40%	2.40%
Deletion	20.10%	8.90%	7.10%
Word Error Rate	7.3%	6.2%	4.40%

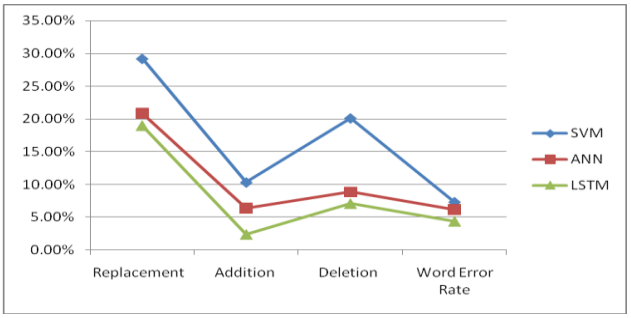

Fig. 6 WER based on insertion, deletion and replacement

VI. CONCLUSIONS

In this paper, we focused on classifying the stuttering and non-stuttering speech signal by using LSTM classifier and then correcting the prolongation and repetition of words in a given stutter speech signal. Corrected signal is stored in database. Finally with the corrected speech signal we predicted corresponding text for a given stutter speech signal and also reduced 2% of WER when compare with SVM and ANN.

REFERENCES

1. R.Klevansand R.Rodman, "Voice Recognition, Artech House, Boston, London 1997.
2. M.A.Anusuya , S.K.Katti "Speech Recognition by Machine: A Review" International journal of computer science and Information Security 2009.
3. M.A.Anusuya and S.K.Katti, "Speech Recognition by Machine: A Review", (IJCSIS) International Journal of Computer Science and Information Security, vol. 6, no. 3, pp. 181-205, 2009
4. Kuldeep Kumar R. K. Aggarwal, "Hindi speech recognition system using HTK", International Journal of Computing and Business Research, vol. 2, issue 2, May 2011.
5. Mohit Dua, R.K.Aggarwal, Virender Kadyan and Shelza Dua, "Punjabi Automatic Speech Recognition Using HTK", IJCSI International Journal of Computer Science Issues, vol. 9, issue 4, no. 1, July 2012.
6. D. Yu and L. Deng, Automatic Speech Recognition—A Deep Learning Approach. New York, NY, USA: Springer, Oct. 2014.
7. M. Hariharan, V. Vijean, C. Y. Fook, and S. Yaacob, "Speech stuttering assessment using sample entropy and Least Square Support Vector Machine," in IEEE 8th International Colloquium on Signal Processing and its Applications (CSPA), 2012, pp.240-245, 23-25 March, 2012.
8. Kaushik, M., Trinkle, M., Hashemi-Sakhtsari, A. 2010. Automatic detection and removal of disfluencies from spontaneous speech. Proc. 13th Australasian Int. Conf. on Speech Science and Technology Melbourne, 98-101.
9. M. Gales, S. Young, " The application of hidden Markov models in speech recognition", Found. Trends Signal Process., 1 (3) (2007), pp. 195-304
10. Nöth, E., Niemann, H., Haderlein, T., Decher, M., Eysholdt, U., Rosanowski, F., et al. (2000). Automatic stuttering recognition using hidden Markov models.
11. L. Helbin T. Tian-Swee and S. H. Salleh. "Application of Malay speech technology in Malay Speech Therapy Assistance Tools". In: Intelligent and Advanced Systems (2007), pp. 330–334.
12. C. Burges, "A tutorial on support vector machines for pattern recognition," Data Mining Knowl. Discov., vol.2, pp. 121-167, 1998.
13. A. Reda, El-Khoribi, "Support Vector Machine Training of HMT Models for Land Cover Image Classification," ICGST-GVIP, vol.8, issue 4, pp. 7-11, December 2008.
14. K. M. Ravikumar, R.Rajagopal, and H.C.Nagaraj, "An Approach for Objective Assessment of Stuttered Speech Using MFCC Features," ICGST International Journal on Digital Signal Processing, DSP, vol. 9, pp. 19-24, 2009.
15. P. S. Savin, P. B. Ramteke and S. G. Koolagudi, "Recognition of Repetition and Prolongation in Stuttered Speech Using ANN," Proc. 3rd International Conference on Advanced Computing, Networking and Informatics, pp. 65–71, 2016.
16. Manpreet Kaur and Amanpreet Kaur. A Review: Different methods of segmenting a continuous speech signal into basic units. International Journal Of Engineering And Computer Science (2013).
17. Ali H, Ahmad N, Zhou X, Iqbal K, Ali SM (2014) DWT features performance analysis for automatic speech recognition of Urdu. SpringerPlus 3(1):204
18. Lim Sin Chee, Ooi Chia Ai, M. Hariharan and Sazali Yaacob, "MFCC based Recognition of Repetitions and Prolongations in Stuttered Speech using K-NN and LDA" Proceedings of 2009 IEEE student conference on Research and Development.
19. Ravikumar, K. M., Rajagopal, R., & Nagaraj, H. C. (2009). An approach for objective assessment of stuttered speech using MFCC features. ICGST International Journal on Digital Signal Processing, DSP, 9(1), 19–24.
20. Lim Sin Chee, Ooi Chia Ai and Sazali Yaacob, "Overview of Automatic stuttering recognition system" International conference on Man-Machine Systems(ICoMMS) October 2009, Batu Ferringhi, penang, Malaysia.