VIRTUAL ASSISTANT FOR DEAF AND DUMB

Sumanti Saini, Shivani Chaudhry, Sapna,

[1,2,3] *Final year student, CSE, IMSEC Ghaziabad, U.P., India*

ABSTRACT

For the Deaf and Dumb community, the use of Information and Communication Technology has increased the ease of life for them. Deaf and dumb people communicate with help of sign language to pass their messages to each other. In this way, they can't express their ideas the exact way, they want. The paper illustrates implementation of STT and TTS technique for deaf and dumb people to make them communicate better. In this paper, we have compared elaborated research work done in the same field which may be helpful in identifying the drawbacks as well as methods to improvise the present technology. This paper presents a detailed methodology used in numerous research work and advancements for the deaf and dumb community.

Keyword: - *STT (speech to text), TTS (text to speech), ASR (Automatic Speech recognition), DNN (Deep Neural Network), HMM (Hidden Markov Model).*

1. INTRODUCTION

At present time, smart phones are quite popular and easily accessible. They have features that can help any human being to make a tedious task rather simpler. If we are able to translate voice input in text in real time, it can bring a drastic change in life of deaf and dumb people. At present, technology has enhanced and advanced itself exponentially. Gestures may have a limit for use as not all the people are familiar with sign language and also the fact that various region in the world uses different sign languages. We are trying to implement a communication mechanism in order to overcome differences between the deaf and dumb community as the normal people. Speech to Text translation allows us to convert perceptible language into text that they can read, edit as well as write on smart phones.

Our primary concern is to avail a way for deaf and dumb community to get closer to the Technical Era by making use of STT and TTS technology. It can be helpful to provide automated voice over for the dumb community. It also provides speech to text translation to interpret the message to the deaf community which in future can be used on internet calls. TTS system can be implemented in various languages. It helps deaf and dumb people to express their feelings and bridge the gap between normal and deaf as well as dumb community. One method to implement Speech to text translation in real time is by making use of HMM i.e. Hidden Markov Model. For text to speech translation, we can use Deep Neural Network. TTS mechanism is implemented in five steps. That is segmentation model that is used to locate boundaries for phonemes, conversion model from grapheme to phoneme, prediction model for phoneme duration and the audio synthetisation model.

2. LITERATURE SURVEY

The paper has illustrated and compared different techniques as GMM-HMM. It only performs in noiseless environment with MFCC features [1] and not really robust. Another Model is DNN-HMM which gives lower word error rate [3]. It is not really robust for many layer and higher cost computational [4]. MLP-HMM also can be used for ASR. MLP-HMM outperforms in clean and noisy environment [1] but it is quite complicated due to MLP does not has any specific rule [1]. SVM-HMM gives higher accuracy [2] but also not suitable for MFCC features [5]. Hidden Markov Models (HMMs) provide least complicated and time-effective frame-work. Therefore, it is a popular choice for almost all latest, large vocabulary continuous speech recognition (LVCSR) systems [6]. Sultana, S.; Akhand, M. A H; Das, P.K.; Hafizur Rahman, M.M. introduces to Speech-to-Text(STT) translation with help of SAPI for Bangla. Since the obtained precision is high enough for TTS comprising research work, there are several components that that can enhance its performance and help to achieve better results that is to implement TTS in other languages than Bangla [7]. Yee-Ling Lu, Man-Wai and

Wan-Chi Siu illustrate the phoneme distribution and phoneme frequency implemented with Recurrent Neural Networks. It is under operated with help of real time recurrent learning (RTRL) algorithm [8].

3. SYSTEM COMPONENTS:

3.1. TTS:

i. **The grapheme-to-phoneme model-** Initially, it converts the written form of text (i.e. English characters) to voice segments that are converted into coded form with help of a phonemic dictionary.

ii. **The segmentation model-** This model draws boundaries for phonemes from the data input. It illustrates the beginning and end of Audio input provided the voice dataset and a phoneme-by-phoneme description of the input files.

iii. **The phoneme duration model** -This duration model basically predicts the temporal duration of each phoneme in an utterance (i.e. phoneme sequence).

iv. **The fundamental frequency model –** This model prognoses if a given phoneme is uttered or not. If it is so, the model is used for forecast of the overall fundamental frequency (F_0) in the phoneme's time interval.

v. **The audio synthesis model-** It collaborates the outcomes of all the previous models and synthesizes the transcripted phoneme sets with respect to the targeted text.

The second phase of text to speech system is inference. In this phase, any of two methodologies is used to take the text as input. Either, text is taken via the grapheme-to-phoneme module or a phoneme dictionary is used to generate phonemes. After this, the phonemes are generated to be given as input to other modules, that are the phoneme duration prediction model and frequency prediction model. These models are used to allocate time intervals of its existence to each phoneme and to forecast a frequency con-tour. Now at the end, local conditioning input features in form of the phonemes, phoneme intervals, and their corresponding frequencies are provided as input to the audio synthesis model.

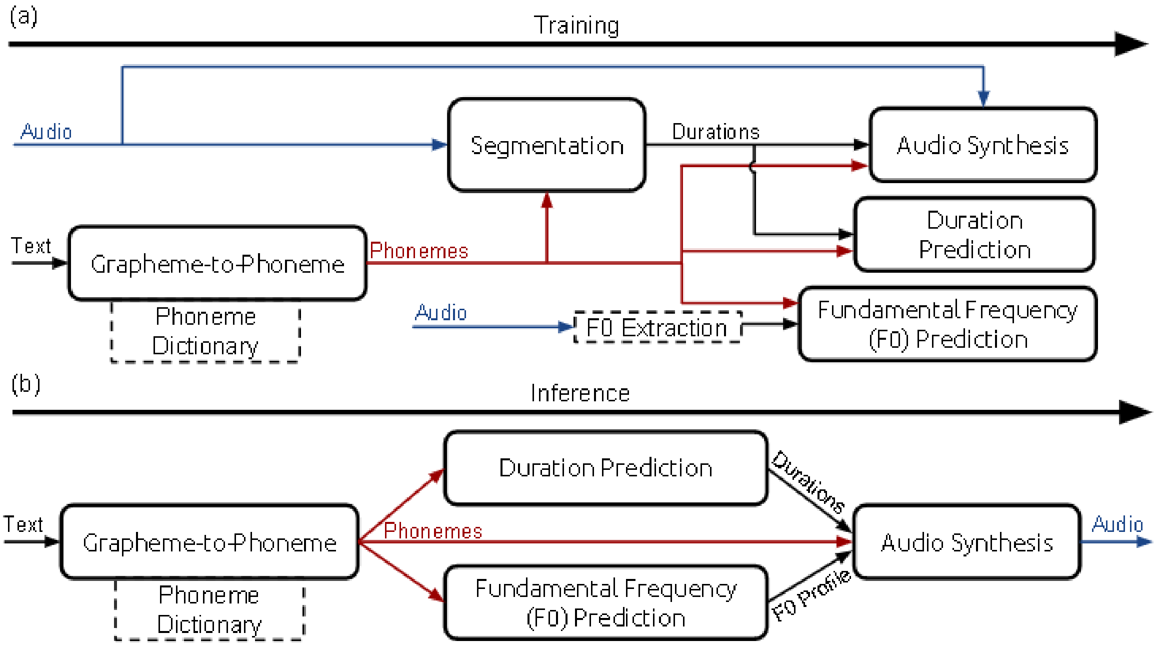

Figure 1. (a) training mechanism (b) Inference mechanism, where input sets are provided on LHS and output sets are provided on RHS. In the given module, time interval i.e. duration as well as frequency is predicted by same Neural Net prepared with a joint loss. Non-learned components are represented by Dotted lines.

It creates the final voice using the phoneme and its attributes. During inference phase all other modules are used but segmentation model. Instead, segmentation model is used to map the voice data set with the phoneme boundaries. These limits are described by the time intervals which are taken reference of, while tutoring the phoneme duration model in first phase. Training of last pass of inference phase is done by the audio marginalized by phoneme set, its interval and F_0.

3.2 STT:

With advanced technology, Speech to text translation is much easier than it was few decades or years back. At present time, STT can compile most of the spoken words exactly and translate over 92+x % of total phoneme in multiple languages. Although, this much accuracy is not enough, as 95+x % is required to process a text and transmit it over network for it to be understandable (Stinson et al. 1999: accuracy).

To achieve even this 92+x% precision in ASR, the trainer has to fore mostly tutorise the modules with given speech samples, which is a tedious task itself. Though, achieving 100% precision is quite difficult as there are continental accents, that are normally very badly recognised in spite of experiencing intensive training by the module. Noise in environment adds difficulty level in recognition process. Mapping of physical parameters with original voice input is another factor for ASR. It is created by the generic module of linguistic as well as the phoneme and the data set from each training and testing result. On the basis of difference of individual physical parameters with results of the training parts, we can conclude the error of STT implementation. One more thing to be taken note of, if environmental disturbance makes the signal-to-noise-ratio reduce, precision maybe effected negatively by 10+x%. Hidden Markov Model is used for Isolated Word recognition. We have extracted MFCC features for a particular wave file of the utterance 'one.wav'. It has nearly 30 frames where each frame contains 13 coefficients. The phonetic representation of one=w ah n. Initial assumption is that each phoneme to be represented by 3 states S1, S2 and S3.

- The first 10 frames (phoneme 'w') will be modelled using 3 states, next 10 frames (phoneme 'ah') using other 3 states and next 10 frames (phoneme 'n') using other 3 states. Thereby each phoneme is represented by 3 states.
- Each phone (consecutive 3 states) is represented by 3*3 transition probability matrix.
- Since there are 3 states, each state is assumed to be represented by single GMM with 4 gaussians. Hence each state will contain Mean vector [4*13], Variance vector [4*13] and priori weights [4*1].
- Each phone is represented by 3 different mean vectors (m1, m2 and m3) where each one each of size 4*13, variance vector (4*13) and Weight vector (4*1). Each phone is represented by a single State transition matrix (3*3) and Emission probability matrix (3*10), 10-frames.

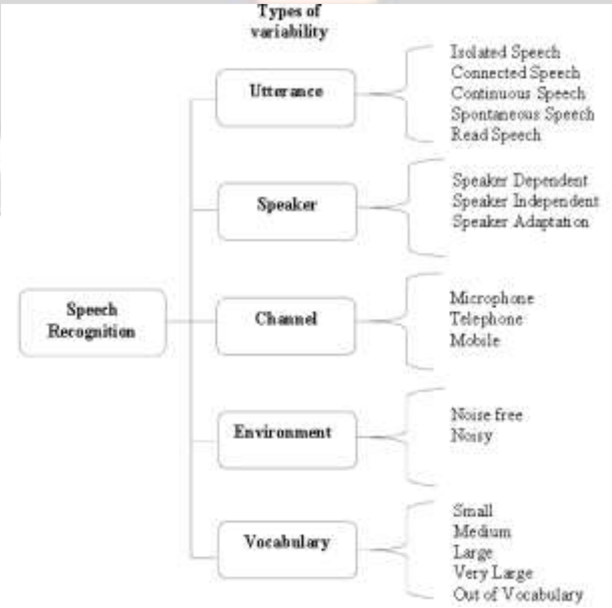

Figure-2, Variations in Speech.

After initialization of HMM parameters, we need to compute the forward probabilities (alpha's) and backward probabilities (beta's) for every phoneme. Further, the state transition matrix and Emission probability matrix for each of the phoneme HMM is updated until there are no significant changes in the log likelihood after successive iterations. After completion of training phase, Viterbi decoding is used to find the hidden state sequence given the observation sequences and the model parameters.

Initially, we need to correctly segment the utterance signal, phonemes don't have the same length. Each ith phoneme is described by definite number of frame Ni. (one=30 frame, N1=12, N2=16, N3=8 for example; although there is some overlapping between phonemes). After segmentation, each phoneme is modelled as triphone since it is not possible to avoid the overlapping between adjacent phonemes, so for the given example of one = w ah n, the model will be one= sil-w-ah, w-ah-n, ah-n-sil. Therefore, a phoneme is again sub-segmented into three phones: beginning, centre, end. Each phone (sub-phoneme) is modelled by a GMM associated with an HMM state (in sil-w-ah for instance, S1/GMM1 will model sil-, S2/GMM2 will model -w- and S3/GMM3 will model -ah). A state has two transition probabilities and one emission probability. Addition of an input and output states is provisional. (if N1=12 for the phoneme sil-w-ah, sub-phoneme length maybe N11=2, N12=8, N13=3 for example, overlapping is again unavoidable). To model a phoneme, we need more than one utterance. In case, there are 50 utterances of the same word, only thing need to be done is to segment them correctly into phonemes and then sub-phonemes (the length of the same phoneme, and therefore sub-phoneme, may vary from one utterance to another). For each sub-phoneme collect frames that are associated to it, model them as a GMM, then train the HMM. GMM can be shared between HMMs.

CONCLUSION:

This project implements TTS using deep Neyral Network and STT using Hidden Markov Model. It provide a module that can be used for real time conversion which can be used to assist deaf and dumb people to transmit their messages. This system is proposed to improve lifestyle of dumb/ deaf person's. This project is also favourable for degrading the communication difference between the blind person and the dumb person.

All over the project is effective and efficient because it is using the TTS modules and further building efficient system for delivery of emotional prosody. This paper is helpful for the industry of people working in the area of designing systems based on Speech synthesis.

FUTURE SCORE:

To outstrip the shortcomings of HMM, an enhanced module can be used that is hybrid HMM -Artificial Neural Network system with ANN. Artificial Neural Network comprises of a Multilayer Perceptron network. Its frame-based outputs represent posterior probabilities of phoneme frequencies and is used as state occupancy probabilities in HMMs. Thus, it is capable of solving much more complicated recognition tasks, and can handle low quality, noisy data, and speaker independence. The artificial neural networks (ANN), on the other hand, though poor in handling time-sequences, have good pattern discriminative power and can incorporate contextual information rather easily.

REFERENCES:

[1] P. Pujol, S. Pol, C. Nadeu, A. Hagen, and H. Bourlard, "Comparison and combination of features in a hybrid HMM/MLP and a HMM/GMM speech recognition system," IEEE Transactions on Speech and Audio processing, vol. 13, pp. 14-22, 2005.

[2] E. Zarrouk, Y. B. Ayed, and F. Gargouri, "Hybrid continuous speech recognition systems by HMM, MLP and SVM: a comparative study," International Journal of Speech Technology, vol. 17, pp. 223-233, 2014.

[3] M. L. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in
Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on, 2013, pp. 7398-7402.

[4] O. Abdel-Hamid, L. Deng, and D. Yu, "Exploring convolutional neural network structures and optimization techniques for speech recognition," in Interspeech, 2013, pp. 3366-3370.

[5] Y. Zheng, "Acoustic modeling and feature selection for speech recognition," Citeseer, 2005.

[6] J. A. Arrowood, Using Observation Uncertainty for Robust Speech Recognition. PhD thesis, Georgia Institute of Technology, 2003.

[7] Deepa V.Jose, Alfateh Mustafa, Sharan R,"A Novel Model for Speech to Text Conversion" International Refereed Journal of Engineering and Science (IRJES)ISSN (Online) 2319-183X,
Volume 3, Issue 1 (January 2014)

[8] Chen, Jingdong, Chrzanowski, Mike, Coates, Adam, Diamos, Greg, et al. Deep speech 2: End-to-end speech recognition in English and mandarin. arXiv preprint arXiv:1512.02595, 2015.

[9] Boersma, Paulus Petrus Gerardus et al. Praat, a system for doing phonetics by computer. Glot international, 5, 2002.

[10] James, Merity, Stephen, Xiong, Caiming, and Socher, Richard. Quasiecurrent neural networks. arXiv preprint arXiv:1611.01576, 2016.

[11] KyungHyun, and Bengio, Yoshua. Empirical evaluation of gated re-current neural networks on sequence modelling arXivpreprint arXiv:1412.3555, 2014.

[12] Dukhan, Marat. Peachpy meets opcodes: direct machine code generation from python. In Proceedings of the 5th Workshop on Python for High performance and Scientific Computing, pp. 3. ACM, 2015.