Blizzard Challenge 2015 : Submission by DONLab, IIT Madras

Anusha Prakash[1], *Arun Baby*[2], *Aswin Shanmugam S*[2], *Jeena J Prakash*[2], *Nishanthi N L*[2], *Raghava Krishnan K*[3], *Rupak Vignesh Swaminathan*[2], *Hema A Murthy*[2]

[1]Dept of Applied Mechanics, Indian Institute of Technology Madras, India
[2]Dept of Computer Science and Engineering, Indian Institute of Technology Madras, India
[3]Dept of Electrical Engineering, Indian Institute of Technology Madras, India
hema@cse.iitm.ac.in

Abstract

As part of Blizzard Challenge 2015, text-to-speech synthesisers have been developed for Indian languages. This paper presents the work done by the DONLab team, IIT Madras for the Challenge. With the provided speech data for six Indian languages, Hidden Markov Model based speech synthesis systems and STRAIGHT voices have been built. Various modules involved in system building have been described. While some modules are language-specific, systems have been built mostly language-independently. Of interest, is the novel hybrid segmentation algorithm to obtain accurate labels at the phone level. Monolingual and multilingual synthesised speech output for the given test sentences have been submitted. In the results of evaluation, "D" is the identifying letter of our systems. Modifications to the training process, post-submission of the synthetic sentences, have also been briefly described.

Index Terms: Blizzard Challenge, Indian languages, Hidden Markov Model, hybrid segmentation

1. Introduction

Blizzard Challenge is an international platform to compare and learn about the latest research techniques in text-to-speech (TTS) synthesis domain. The Challenge for the year 2015 consists of the Main tasks and the Pilot task. The Main tasks comprise of building Indian speech synthesisers for six Indian languages - Bengali, Hindi, Malayalam, Marathi, Tamil and Telugu. The Main tasks are further divided into two - Hub task, which involves building a voice in each language; and Spoke task, for which multilingual sentences containing words in English and the native language have to be synthesised. The Pilot task consists of developing an English TTS from the given data. In this paper, only the efforts towards completing the Main tasks have been presented.

The given languages can be classified into two language groups: Indo-Aryan and Dravidian. Bengali, Marathi and Hindi are Indo-Aryan languages, while Malayalam, Tamil and Telugu are Dravidian languages. Owing to the similarities among the languages, a language-independent system building approach has been adopted. The common label set and the common question set have been used for this purpose [1]. Letter to sound rules have been carefully hand-crafted for one Indo-Aryan and one Dravidian language, and extended to other languages in the group.

Accurate segmentation of data is required for building a good quality TTS. Phone level labels have been obtained using a hybrid segmentation algorithm [2]. The novelty of this technique is that it uses signal processing concepts in tandem with machine learning algorithms to achieve accurate segmentation. This technique has been used to segment data of all languages. For Hindi and Tamil, an improved version of the hybrid segmentation algorithm has been used.

Phone based Hidden Markov Model speech synthesis systems (HTS) [3] along with STRAIGHT systems [4] have been built for every language. The system yielding the best synthesis output among the two have been chosen based on informal listening tests. The details about the same are mentioned in Table 1.

Table 1: Systems built for different languages

Language	System built
Bengali	HTS+STRAIGHT
Hindi	HTS+STRAIGHT
Malayalam	HTS
Marathi	HTS+STRAIGHT
Tamil	HTS
Telugu	HTS+STRAIGHT

So far, the Hub task has been described. For the Spoke task, multilingual test sentences need to be synthesised. These sentences contain words in English and the native script. The systems built for the Hub task are also used here. The challenge is in handling English words. A classification and regression tree (CART) has been developed for the same.

Some efforts have gone into improving the quality of the synthesis speech output. This work has been undertaken after the submission of synthetic sentences. The improved hybrid segmentation algorithm has been used to segment data for all the six languages. Next, the database has been pruned using a pruning algorithm and this pruned database has been used to build models for the HTS framework. Results of subjective evaluations comparing the two versions of TTSes have been presented.

The rest of the paper is organised as follows. Section 2 describes the speech data provided for the Challenge. The procedure to build Indian language synthesisers are detailed in Section 3. The parsing rules for the languages are mentioned. The hybrid segmentation algorithm is also explained briefly. Section 4 describes how English words have been tackled to synthesise multilingual sentences. In Section 5, the results of system evaluations are discussed. In Section 6, a brief description of work done post-submission of synthetic sentences is presented.

2. Speech Database

Speech data, consisting of wave files with the corresponding text, has been provided for six Indian languages. Table 2 mentions the duration of data for each language. Speech of native professional speakers has been recorded at a sampling rate of 16 kHz. Correct transcriptions have been provided in the corresponding native script in UTF-8 format, with commas marked appropriately wherever the speaker has paused.

Table 2: Duration of data provided

Language	Duration (Hours)
Bengali	2
Hindi	4
Malayalam	2
Marathi	2.2
Tamil	4.3
Telugu	4.2

3. Training Indian Speech Synthesisers

The flowchart of developing a TTS along with the synthesis procedure is shown in Figure 1. In the training process, the training text is parsed into a set of labels representing monophones. Wave files are segmented at the phoneme level using the automatic hybrid segmentation algorithm [2]. HTS and HTS+STRAIGHT voices are built. During synthesis, the test sentence is checked for English words. English words are parsed using a classification and regression tree (CART) and native words using the native script parser.

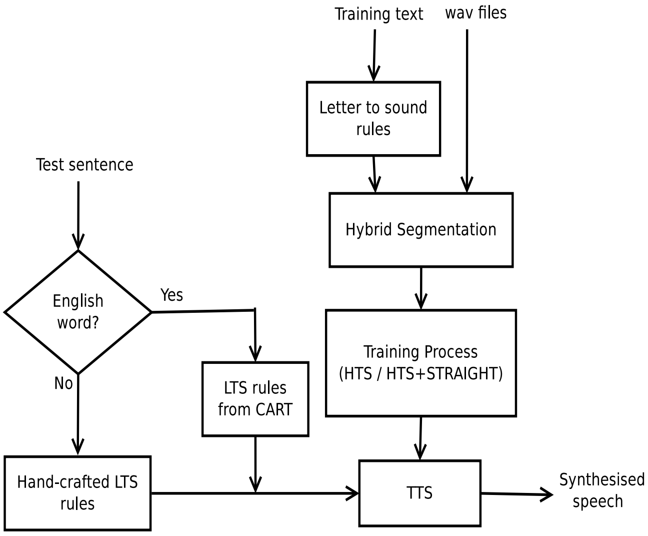


Figure 1: Flowchart of voice building and text synthesis

3.1. Letter to sound (LTS) rules

A set of hand-written rules have been developed for the letter to sound correspondence. The grapheme to phoneme mapping is mostly one-to-one in Indian languages. Some language specific rules have been included for different languages. Very specific parsers have been developed for Hindi and Tamil. In Hindi, *schwa* (ə) deletion has been handled. This is the deletion of the short vowel /a/ at the end or at the middle of a word. For example, the grapheme representation of the Hindi word **/kaamcor/** is /k aa m a c o r a/, while its phoneme representation is /k aa m c o r/. A standard set of rules has been applied to handle this. Parsers for Marathi and Bengali, which are Indo-Aryan languages like Hindi, have been derived from Hindi.

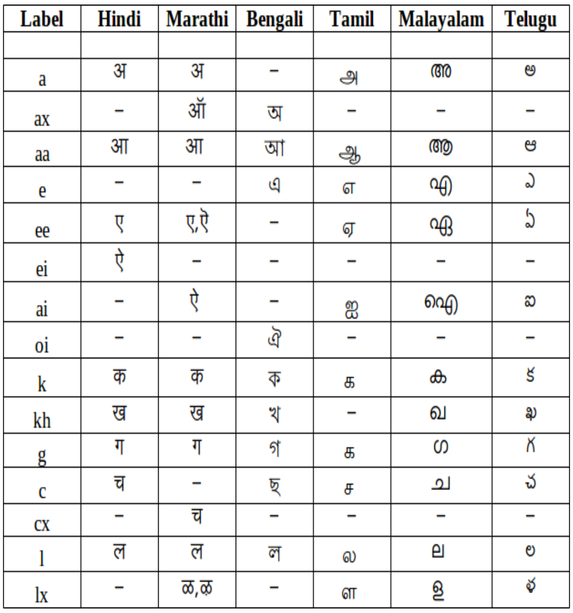
The parsing in Tamil is simpler. Tamil consists of a smaller set of sounds. The same character represents an unvoiced stop consonant and its corresponding voiced counterpart. The pronunciation depends on the place and manner of articulation of the surrounding phonemes. There are a standard set of rules for this. But we denote the character by a single label, as modeling takes care of this contextual variation. The Tamil parser has been extended to accommodate the larger set of characters for Malayalam and Telugu.

Additionally, Malayalam has some special characters called *chillus* or *chillaksharas*. These are characters that represent pure consonants and are never followed by vowels. They do not appear in the beginning of a word. *Chillaksharas* have been denoted by separate labels.

Hybrid segmentation needs syllable labels as input. Hence syllable parsers have been developed for the languages based on LTS rules. The native word is first syllabified and then phonified.

3.2. Common label set

There are some common sounds present across the six languages. They are mapped together and denoted by a single label. Language specific sounds are denoted by different labels. This set of labels is the common label set [1]. The output of the parser is in terms of these labels. The partial set is shown in Figure 2. This standard notation across languages aids in building systems language-independently.



Label	Hindi	Marathi	Bengali	Tamil	Malayalam	Telugu
a	अ	अ	–	அ	അ	అ
ax	–	ऑ	অ	–	–	–
aa	आ	आ	আ	ஆ	ആ	ఆ
e	–	–	এ	எ	എ	ఎ
ee	ए	ए,ऐ	–	ஏ	ഏ	ఏ
ei	ऐ	–	–	–	–	–
ai	–	ऐ	–	ஐ	ഐ	ఐ
oi	–	–	ঐ	–	–	–
k	क	क	ক	க	ക	క
kh	ख	ख	খ	–	ഖ	ఖ
g	ग	ग	গ	க	ഗ	గ
c	च	–	ছ	ச	ച	చ
cx	–	च	–	–	–	–
l	ल	ल	ল	ல	ല	ల
lx	–	ळ,ऴ	–	ள	ള	ళ

Figure 2: Partial common label set

3.3. Hybrid segmentation algorithm

Segmentation of speech into phonemes is a cardinal task in building TTS systems. The HMM-based segmentation exploits the knowledge of sequence of phonemes and trains phoneme models using Baum-Welch reestimation. During forced Viterbi alignment, the sequence of most likely phone states within frames is used for deriving boundaries and the actual acoustic landmarks are missed. Hence, boundaries are not represented by this model. On the other hand, the group delay based segmentation smoothes the short-term energy (STE) function by making use of the additive property of Fourier transform phase and deconvolution property of the cepstrum, thereby deriving syllable boundaries from the smoothed STE function [5].

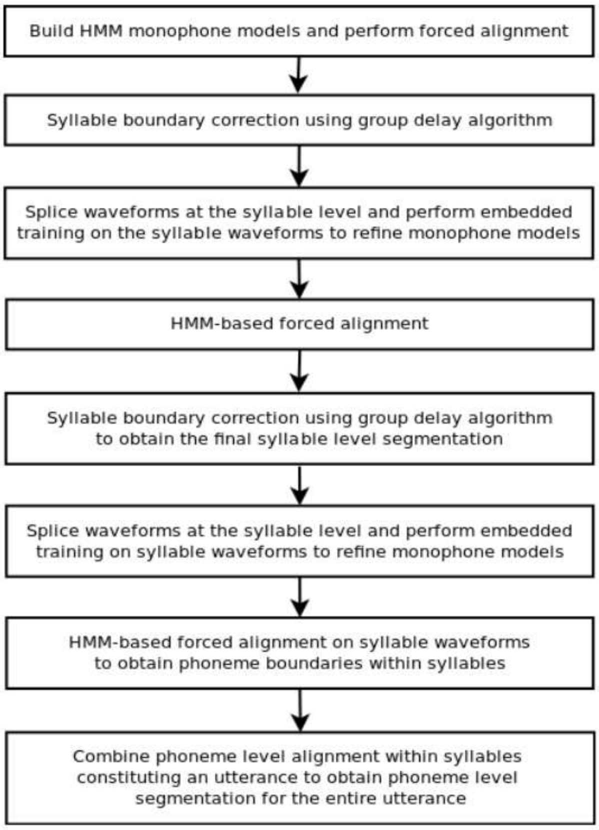

Build HMM monophone models and perform forced alignment
Syllable boundary correction using group delay algorithm
Splice waveforms at the syllable level and perform embedded training on the syllable waveforms to refine monophone models
HMM-based forced alignment
Syllable boundary correction using group delay algorithm to obtain the final syllable level segmentation
Splice waveforms at the syllable level and perform embedded training on syllable waveforms to refine monophone models
HMM-based forced alignment on syllable waveforms to obtain phoneme boundaries within syllables
Combine phoneme level alignment within syllables constituting an utterance to obtain phoneme level segmentation for the entire utterance

Figure 3: Flowchart of hybrid segmentation

Indian languages being phonetic, predominantly have straight forward syllabification rules. The consistent coordination between the acoustical and the lexical units, make it possible to combine these two different segmentation procedures under a common framework [6] by restricting the Baum-Welch reestimation procedure within the syllable boundaries dictated by group delay algorithm. The phoneme boundaries within these syllables are refined when forced alignment is performed at the syllable level. Since the boundaries given by the group delay algorithm are dependent on the size of the lifter on the root cepstrum, it may suffer from insertions and deletions. Previously, a semi automatic labeling tool was used to manually correct the syllable boundaries [7]. Later, the process was automated by choosing the lifter size such that it always allow insertions and the syllable boundaries given by HMMs were approximated to only boundaries of high confidence [2]. For Hindi and Tamil systems, an additional cue known as spectral flux was used in order to correct boundaries that could not be corrected by smoothed STE function [8]. The spectral flux is a function which quantifies change in spectral content over time. The syllable boundaries that were characterized by an abrupt change in spectral energy were corrected by spectral flux [8].

3.4. Common question set

All the labels in the common label set across the six languages have been included in the question set. This common question set [1] has been used for tree based clustering in order to build HTS voices.

4. Synthesising Multilingual Sentences

In order to build letter to sound (LTS) rules for bilingual TTS, we employed the following procedure. A word list containing about 6000 English words was formed. The list included words from the CMU Arctic database which is phonetically balanced and also words from the Chandamama database. These words were first transliterated to Indian languages using an online transliteration tool. After the transliteration, a native speaker was asked to verify the correctness. These words were then parsed by the monophone parsers of the respective languages to create a dictionary of words with their corresponding phone level transcriptions. This dictionary was used to build the classification and regression tree [9]. LTS rules for parsing English words are derived from CART.

5. Evaluation of Systems and Discussion

The Hub task consists of synthesising monolingual test sentences. There are two types of test sentences - read text (RD) and semantically unpredictable sentences (SUS), each 50 in number. For the Spoke task, 50 multilingual sentences have to be synthesised. The multilingual test sentences consist of words in English and the native script.

Synthesised sentences were evaluated by a set of listeners well-versed in the Indian language. They were asked to rate the audio files based on similarity to the original speaker and naturalness of the synthesised speech. Word error rate (WER) was calculated from a set of transcribed sentences, as a measure of the intelligibility of the system. Some of the results of the evaluation are discussed here. Overall 8 systems have been submitted for Hub task and 5 for the Spoke task. In all the plots, “A” corresponds to natural (recorded) sentences and “B” is the baseline system. The identifying letter for our system is “D”.

Results of the Hub task are discussed first. The performance of our system is better than the baseline system in most cases. The similarity to original speaker and naturalness for read text are same for almost all languages. While naturalness of SUS is rated higher for Hindi, Malayalam and Tamil, similarity to the original speaker is better for Marathi and Telugu. For Bengali, similarity and naturalness of SUS have been rated the same scores. Another observation is that the scores for our systems are spread across a wide range, indicating that the the synthesis quality is better for some test sentences compared to others and not uniform throughout.

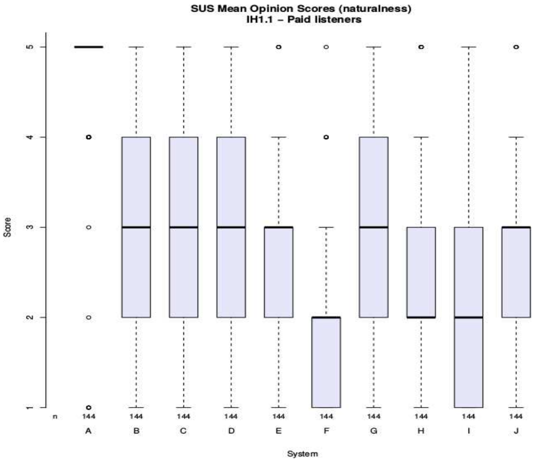


Figure 4: Naturalness score for Bengali SUS

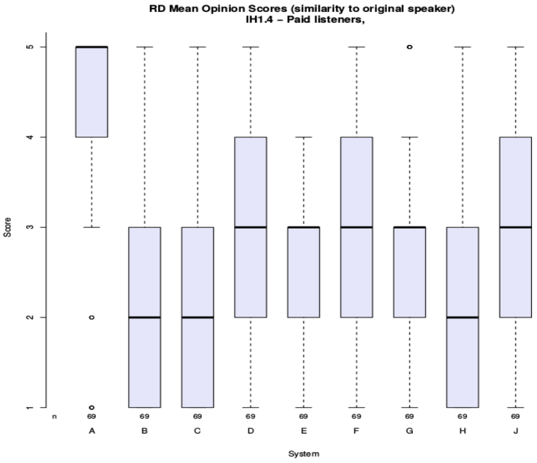


Figure 5: Similarity score for Marathi RD

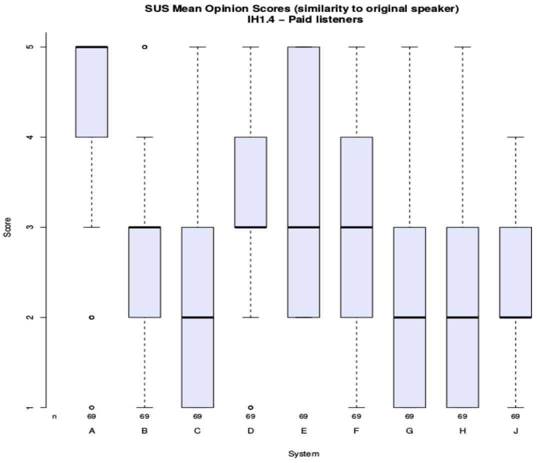


Figure 6: Naturalness score for Marathi SUS

Compared to systems of other teams, the overall performance of Bengali and Marathi systems (Figures 4, 5, 6) have been good, Hindi and Telugu systems average, and Malayalam and Tamil (Figure 7) systems poor. Average WER for our systems is about 54.5%. Hindi has the lowest WER among our systems (Figure 8) and Malayalam the highest (Figure 9). In some cases it can be observed that WER of natural sentences is also high, highest in the case of Telugu as seen from Figure 10. This might be due to spelling errors, agglutinative nature of Indian script, etc, as a result of which automatic comparison increases word error.

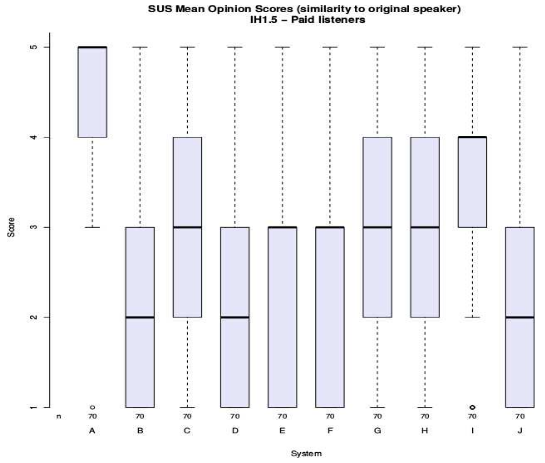


Figure 7: Similarity score for Tamil SUS

Evaluation of Spoke task involves rating the multilingual sentences (ML) based on similarity to original speaker and naturalness. Marathi system has performed the best compared to systems of other languages (Figure 11), and Tamil the worst (Figure 12). The quality of synthesis, especially the articulation of English words, largely depends on the transliteration and building of CART.

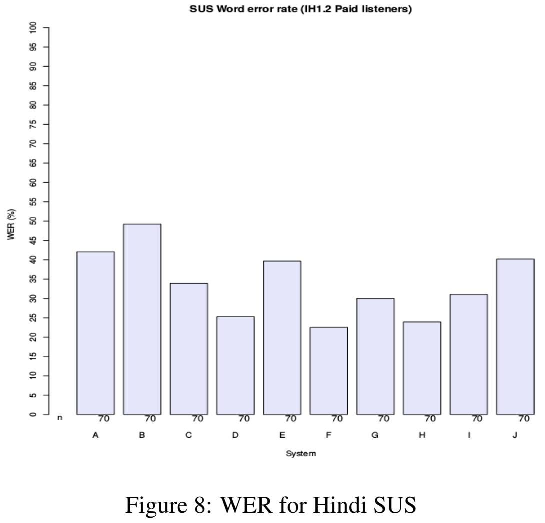


Figure 8: WER for Hindi SUS

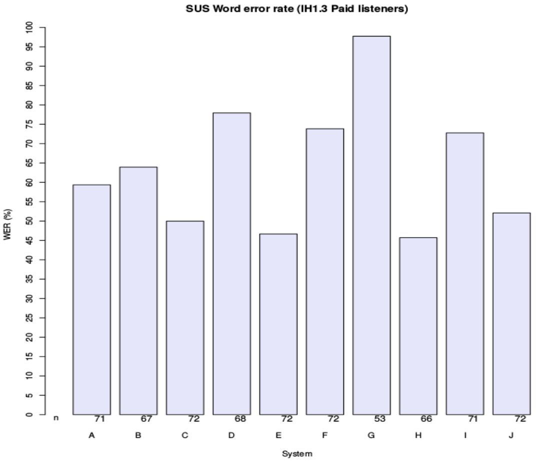


Figure 9: WER for Malayalam SUS

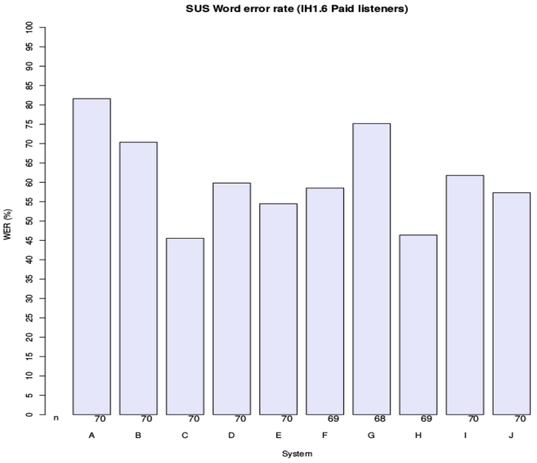


Figure 10: WER for Telugu SUS

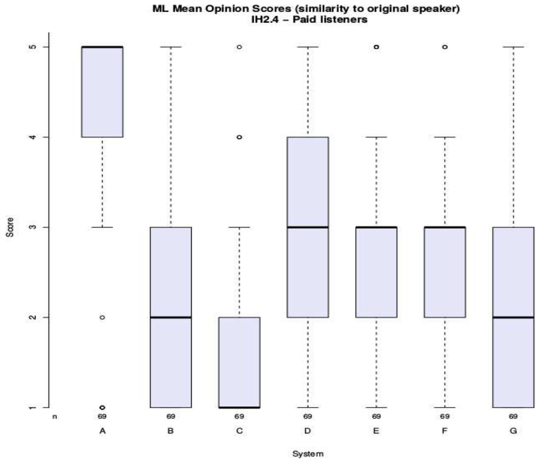


Figure 11: Similarity score for Marathi ML

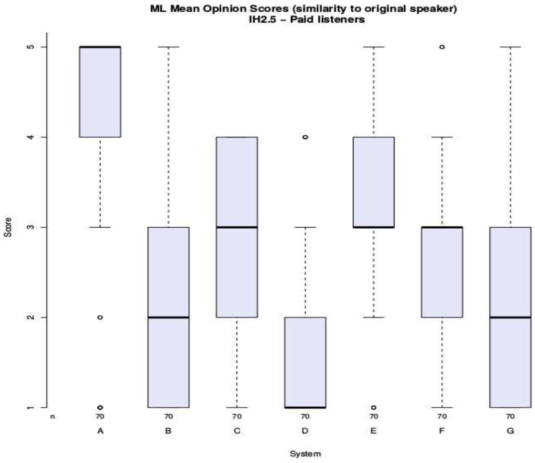


Figure 12: Similarity score for Tamil ML

6. Post Submission Work

After submission of synthetic sentences, data of all languages have been segmented using the new hybrid segmentation algorithm and voices have been re-built. A pruning technique has then been performed [10]. This has been performed mainly for two reasons: (i) since data is recorded mostly over many sessions, there might be variations in the data, (ii) to remove segmentation errors.

The steps for pruning are as follows:

- Syllable level segmentation is obtained from hybrid segmentation algorithm.
- Syllables are tagged separately with positional context. The tags are beg, mid and end, based on the syllable's position in the beginning, middle and end of the word, respectively. The acoustic properties of syllables vary depending on the word positional context [11]. Hence the same syllable with different positional tags is treated as different syllables.
- In addition to positional tags, syllables in Tamil are also tagged with geminate context.
- The duration, average $f0$ and average energy are computed for each syllable using all examples of that syllable.
- The means and standard deviations (σ) of duration, average $f0$ and average energy for each of these syllables are then computed.
- The syllables lying outside 0.25σ are tagged with a special symbol such that they will not be used for building models.
- Only syllables with greater than 3 occurrences in the database are chosen for pruning.
- If there are greater than 50 occurrences of the syllable after pruning, the first 50 occurrences are retained.
- Only the monophones constituting the syllables retained after pruning are used for building monophone models.
- These models are then used as initial models to segment the entire data at the phone level.

A pairwise comparison test [12] has been conducted to evaluate the quality of the synthesised sentences of the new voices compared to that of the submitted audio files. About 8 native listeners have undertaken the test for each language. For Bengali, Hindi, Tamil and Telugu, evaluators have given equal preference to both versions of the systems. However, for Malayalam and Marathi, evaluators have preferred the later version. The results for these two languages have been presented in Table 3.

Table 3: Results of pairwise comparison test

Language	**A-B**	**B-A**	**A-B + B-A**
Malayalam	54.28%	20%	67.14%
Marathi	93.33%	40%	76.67%

7. Acknowledgements

The authors would like to thank the organisors of Blizzard Challenge for this opportunity to interact with other research teams. The authors would also like to thank the Department of Information Technology, Ministry of Communication and Technology, Government of India for funding the project "Development of Text-to-Speech Synthesis for Indian Languages Phase II" (Ref. no. 11(7)/2011-HCC(TDIL)), as part of which, most of the research techniques have been developed.

8. References

[1] B. Ramani, S. Lilly Christina, G. Anushiya Rachel, V. Sherlin Solomi, M. K. Nandwana, A. Prakash, S. Aswin Shanmugam, R. Krishnan, S. Kishore, K. Samudravijaya, P. Vijayalakshmi, T. Nagarajan, and H. A. Murthy, "A common attribute based unified HTS framework for speech synthesis in Indian languages," in *SSW8*, Barcelona, Spain, 2013, pp. 291–296.

[2] S. A. Shanmugam and H. Murthy, "A hybrid approach to segmentation of speech using group delay processing and HMM based embedded reestimation," in *INTERSPEECH*, Singapore, 2014.

[3] H. Zen, K. Tokuda, and A. W. Black, "Statistical parametric speech synthesis," *Speech Communication*, vol. 51, no. 3, pp. 1039–1064, November 2009.

[4] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveign, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," *Speech Communication*, vol. 27, no. 34, pp. 187 – 207, 1999.

[5] V. K. Prasad, T. Nagarajan, and H. A. Murthy, "Automatic segmentation of continuous speech using minimum phase group delay functions," in *Speech Communication*, vol. 42, no. 3, 2004, pp. 429–446.

[6] S. Aswin Shanmugam and H. A. Murthy, "Group delay based phone segmentation for HTS," in *National Conference on Communication (NCC)*, Kanpur, India, February 2014, pp. 1–6.

[7] P. Deivapalan, M. Jha, R. Guttikonda, and H. A. Murthy, "DONLabel: An automatic labeling tool for Indian languages," in *National Conference on Communication (NCC)*, IIT Bombay, India, February 2008, pp. 263–266.

[8] S. Aswin Shanmugam, "A hybrid approach to segmentation of speech using signal processing cues and Hidden Markov Models," http://lantana.tenet.res.in/thesis.php, M S Thesis, Department of Computer Science and Engineering, IIT Madras, India, July 2015.

[9] A. W. Black, K. A. Lenzo, and V. Pagel, "Issues in building general letter to sound rules," in *SSW'98*, 1998, pp. 77–80.

[10] K. Raghava Krishnan, "Prosodic analysis of Indian languages and its application to text to speech synthesis," http://lantana.tenet.res.in/thesis.php, M S Thesis, Department of Electrical Engineering, IIT Madras, India, July 2015.

[11] Venugopalakrishna.Y.R., Vinodh.M.V., H. A. Murthy, and C. Ramalingam, "Methods for improving the quality of syllable based speech synthesis," in *Proc. of Spoken Language Technology (SLT) 2008 workshop*, Goa, India, December 2008, pp. 29 –32.

[12] P. Salza, E. Foti, L. Nebbia, and M. Oreglia, "MOS and pair comparison combined methods for quality evaluation of text to speech systems," in *Acta Acustica*, vol. 82, 1996, pp. 650–656.