IIIT Hyderabad's submission to the Blizzard Challenge 2015

Sai Krishna Rallabandi, Anandaswarup Vadapalli, Sivanand Achanta and Suryakanth V Gangashetty

International Institute of Information Technology- Hyderabad, India

{saikrishna.r, sivanand.a, anandaswarup.vadapalli}@research.iiit.ac.in, svg@iiit.ac.in

Abstract

This paper introduces the speech synthesis system developed by IIIT Hyderabad for Blizzard Challenge 2015. Six Indian languages were evaluated this year: Bengali, Hindi, Marathi, Malayalam, Tamil and Telugu. Two tasks were announced for these languages: the monolingual task (IH1 hub task) and the multi-lingual task (IH2 spoken task). We submitted unit selection synthesis systems to both tasks in all languages. Sentence level viterbi search is used to select the reliable speech units among a set of candidate units based on continuity metrics followed by signal correlation based overlap addition method for the concatenation of the selected units.

Index Terms: Text to Speech Synthesis, Speech synthesis, Unit Selection, Blizzard 2015.

1. Introduction

This is the first entry of IIIT-H to Blizzard challenge. Our main aim is to build stable synthesis systems for Indian languages which are both robust and unrestricted in nature. The Blizzard Challenges [1, 2, 3, 4, 5, 6] are an evaluation to compare research techniques across the world for building corpus-based text to speech systems (TTS). The challenge has been extended to Indian languages in 2013[7], involving two sets of tasks - the Hub tasks, to build native language speech synthesisers and the Spoke tasks, to build bilingual (native language and English) systems. The focus of the 2015 Challenge was on the languages Bengali, Hindi, Malayalam, Marathi, Tamil and Telugu. The IIIT Hyderabad team participated in both the tasks. The given test sentences for synthesizing were of 3 kinds; reading sentences (RD), semantically unpredictable sentences (SUS) and sentences (ML) containing interspersed English words.

We have submitted syllable based unit selection systems (USS). Unit selection based speech synthesis systems [8, 9, 10, 11] have become popular due to their highly natural-sounding synthetic speech. These systems have large speech databases containing many instances of each speech unit, with a varied and natural distribution of prosodic and spectral characteristics. When synthesizing an utterance, the selection of the best unit sequence from the database is based on a combination of two costs: target cost(how closely a candidate units in the inventory match the required targets) and join cost(how well the neighbouring units are feasible for joining) [12].

In section 2, we give an overview of the framework we employed for the submission, followed by the evaluation and discussion in section 3 and future scope in section 4.

2. Overview of the Framework

In this section, we give a brief overview of the IIIT-H synthesis system. The framework follows a front-end/back-end architecture with Natural Language processing as front end and Digital signal processing module as back end.

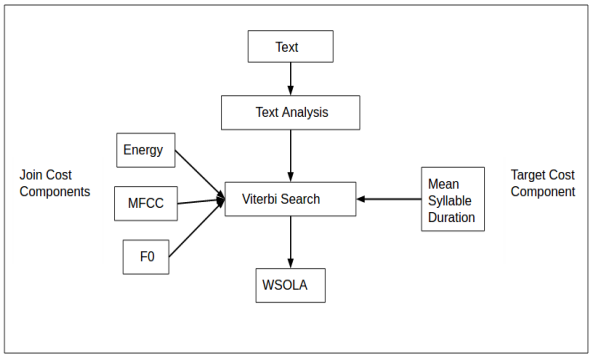


Figure 1: Overview of the Framework

The front end deals with the conversion of natural language text to a structured linguistic representation. This module predicts a sequence of segments called target segments from the raw text. Furthermore, it provides all the essential information regarding prosody. It is composed of a word- and sentence- tokenization module, a text normalizer, a letter-to-sound module and a prosody generator.

The DSP component comprises of the unit selection module and the signal processing module, which relies on a signal correlation based Overlap Add method for speech manipulation. The selection of speech units is performed from the speech database using explicit continuity criteria.

The following are the contributions of the framework in terms of building a unit selection and concatenation

system:

- Integrating a suitable backoff strategy for missing syllables.
- Developing a word to phone mapper for english words to be used in the synthesis of code-mixed(ML) sentences.
- Formulation of a signal correlation based Overlap Addition technique for concatenation of the selected units.

3. System Details

In this section we briefly discuss various modules in the synthesis framework.

3.1. Unit Size

Languages with a very well defined, and a small number of syllables may benefit from a syllable sized unit. Earlier work on Indian languages [13] suggested that a syllable based approach to synthesis could lead to more reliable quality. As Indian languages have a much more regular syllable structure than English we wanted to experiment to find the optimal sized unit for synthesis. The syllable, is of the form V, VC*, C*V and C*VC* where V is the vowel and C is the consonant. The following are the advantages of choosing syllables as the basic unit:

- Syllable units can capture coarticulation better than phonemes, being longer.
- The number of concatenation points decreases when syllable is used as the basic unit.
- Syllables being natural units of production , syllable boundaries are characterized by regions of low energy. Spectral mismatches at the boundary are hardly perceived, provided the inter-syllable pause is preserved.

3.2. Automatic Segmentation

For segmenting the audio data we used the procedure described in [14] which is based on an HMM forced alignment algorithm. The alignment has been performed without any change or supervision as it closely developed to the TTS front-end component. For the current submission, we have not done any kind of bounday correction on top of the obtained labels.

3.3. Syllabification

In Indian languages, words could be composed of basic characters (example samay [time]), as well as complex clusters of C*VC* (example sansthaa [organization]). For the latter cases, there is a need to come up with rules to break the word into syllables. We used the simplistic rules for syllabification as mentioned in [15]i.e. rules for grouping clusters of C*VC* based on heuristic analysis on several words. Apart from this, we have not done any language specific finetuning of the rules.

3.4. Preclustering the units

In Indian languages, poly-syllabic words are prevalent. It was seen that syllables of the same type can be easily differentiated depending on their position in the word [16]. In addition, syllables occurring at the beginning of the word are of longer duration than the syllables occurring at the middle and end of a word [17, 18]. The energy and pitch were also found to vary depending on the position of the syllable in the word [19]. Therefore, we've performed pre-clustering based on position of the syllable in a word, i.e syllables of the same type were pre-clustered as begin, middle and end by appropriately depending on their position in the word, in the original context.

In case a syllable of appropriate position is not available during synthesis, an order of preference is used to pick a syllable of the same type occurring at an alternate position. During synthesis, if the required begin or an end syllable is not present in the database, middle syllable is preferred. If the required middle syllable is not present, a syllable from a word beginning is selected instead.

3.5. Acoustic Features

Typically mel frequency cepstral coefficients (MFCC) are used to calculate distance between two units accompanied by duration and F0 of the unit. Preliminary analysis on the data showed that the energy of units play a major role in syllable unit synthesis. We've therefore included log energy, MFCC, dynamic features of MFCC (deltas and double deltas), F0 and unit durations as the acoustic features.

3.6. Target Cost

We employed a target cost based on the distance from the mean duration of the syllables in the current version of the framework, following [20]. The mean duration for each of the units is computed using all the occurrences in the database. Thus, the units with minimum distance from this mean value have a higher probability in getting selected when the total cost is obtained.

We've also investigated median and the maximum durations as the target cost, both individually and as weighted components in addition to mean duration, but there was no significant improvement.

3.7. Join Cost

Join costs measure spectral and F0 continuity between adjacent units. The subcosts of concatenation cost arise broadly from log energy, spectral and pitch based features. F0 continuity measures were explored in[21, 22, 23]. We use the formulation similar to the one proposed in [21] and use the spectral, f0 and energy based continuity metrics to calculate the concatenation sub costs as

$$C_j^c = \sqrt{\sum_{n=-k}^{k} d_{i,i+1}} \quad (1)$$

where

$$d_{i,i+1} = \left| p_i(k) - p_{i+1}(k) \right|^2$$

$d_{i,i+1}$ is the euclidian distance and $p_i(k)$ is the average pitch value of the kth frame from the ith unit concatenation boundary.

K is the number of frames employed on either side of the concatenation boundary. The value K=0 represents the matching based only on the frames at the concatenation boundary. Value of K is limited by the duration of the available subword unit. Following the observation from [21], we've used the value of k=4.

3.8. Back off Strategy

Syllable based synthesis systems are limited by the syllable coverage in the audio database and missing syllable units are a common occurrence. Back-off methods have been proposed to tackle these issues which either substitute the missing syllable with other syllables in the database or synthesize it using smaller sub word units. However it takes considerable effort to design the substitution rules, requiring detailed perceptual studies.

We use a rule-based back-off method motivated from a perceptual and speech production phenomenon, known as reduced vowel epenthesis, to deal with the missing syllable units. The back off method emulates native speaker intuition in synthesis of the missing units. The observation is that the native speakers of the language break the consonant clusters through vowel insertion to conform to the phonotactics of the language. This phenomenon is known as vowel epenthesis. e.g., The English word bulb which is pronounced by Telugu speakers trained in English as [balb], is pronounced as [balubu] by native Telugu speakers untrained in English. As the consonant cluster "lb" is new to native Telugu speakers, they perform an insertion of the vowel "u" to break it. Another "u" is also inserted after the word final stop consonant b, as words in Telugu do not end with stop consonants. These inserted vowels are called epenthetic vowels. It is this property of epenthesis that we want to exploit as a back-off strategy in Telugu TTS systems. The idea here is to use reduced vowel insertion in complex consonant clusters to replace missing units. The inserted vowel identity is determined using a rule-set adapted from L2 (second language) acquisition research, and are presented in Tables 1 and 2.

The system includes reduced vowel epenthesis as proposed in[24] for Telugu and the same has been extended to the other languages as well.

3.9. Multilingual Synthesis

An informal analysis of a Telugu blog on the web showed that around 20-30 percent of the text is in English (ASCII). Due to the growth of such code mixing it has become necessary to develop strategies for dealing with such multilingual text in TTS systems.

Table 1: Identity of the word medial epenthetic vowel

Following Vowel	Epenthetic Vowel
a,a:	a
i,i:,e,e:	i
u,u:; o,o:	u

Table 2: Identity of the word final epenthetic vowel

Word Final Consonant	Epenthetic Vowel
Non palatal consonant	u
Palatal consonant	i

Following [25], we develop a word to phone mapping to get the phones for the English words. Specifically, we investigate methods of pronouncing English words using Telugu phoneset in the context of Telugu Text-to-Speech.

Our motivation for doing so, comes from our understanding of how humans pronounce foreign words while speaking. The speaker maps the foreign words to a sequence of phones of his/her native language while pronouncing that foreign word. For example, a native speaker of Telugu, while pronouncing an English word, mentally maps the English word to a sequence of Telugu phones as opposed to simply substituting English phones with the corresponding Telugu phones. Hence, we hypothesized that approximating an English word using Telugu phone sequence may be more acceptable for a Telugu native speaker, which is also backed by the case study in [25].

We therefore employed a method of automatically generating word to phone mapping from data for the English words. Letter to phone mapping is not a one to one mapping as each letter may have a correspondence with one or more than one phones, or it may not have correspondence with any phone. As a fixed sized learning vector is required to build a model for learning word to phone mapping rules, we need to align the letter (grapheme) and phone sequences. For this we use the automatic epsilon scattering method, following [26].

The idea in automatic epsilon scattering is to estimate the probabilities for one letter (grapheme) G to match with one phone P, and then use string alignment to introduce epsilons maximizing the probability of the words alignment path. Once the all the words have been aligned, the association probability is calculated again and so on until convergence. This is done in an expectation-maximization based approach and we have followed the same procedure as mentioned in [25].

3.10. Concatenation based on waveform similarity

We use an overlap addition based approach for smoothing the join at the concatenation boundaries. Specifically, we use cross correlation formulation of WSOLA [27].

We've reformulated the algorithm so as to first find a suitable temporal point for joining the units at the boundary. This is done so that the concatenation is performed at a point where maximal similarity exists between the units. In other words, we try to ensure that sufficient signal continuity exists at the concatenation point. For this, we use the cross correlation between the units as a measure of similarity between the units. Then, the units are concatenated at the point of best correlation using cross-fade technique[28] to further remove the phase discontinuities. The number of frames used to calculate the correlation is limited by the duration of the available sub-word unit. In the current framework, we've used the last two frames of the individual units to calculate the cross correlation.

4. Evaluations

This section discusses the evaluation results of our systems. Among all the systems, E is the identifier of our entry. A is the natural speech. We submitted entries of all 6 languages. Similar to previous Blizzard Challenges, in this years challenge three main aspects were put into evaluation via listening tests: naturalness, similarity to the original speaker and word error rate in SUS sentences. The listening subjects were native speakers for each language as each listener had to go through a language dependent CAPTCHA test in order to complete a task for a specific language. To evaluate intelligibility, the subjects were asked to transcribe semantically unpredictable sentences by typing in the sentence they heard; the average word error rates (WER) were calculated from these transcripts. Furthermore, to evaluate the similarity and naturalness, 5-point mean opinion score (MOS) tests were conducted. The scale for the similarity was 5 for sounds like exactly the same person and 1 for sounds like a totally different person compared with a few natural example sentences from the reference speaker. The scale for the naturalness was 5 for completely natural and 1 for completely unnatural.

4.1. Naturalness

We now consider mean opinion scores for naturalness from all listeners on RD and ML sentences. With respect to RD, in two of the languages(Bengali and Tamil), our system has the best performance while in Marathi, our system is outperformed by only two other systems.Overall, our system outperforms between 1 and 4 other systems in each language.With respect to ML, our system has the best performance in every language.

4.2. Speaker Similarity

We now consider mean opinion scores for naturalness from all listeners on RD sentences. In Bengali, we have outperformed every other system. We have comparable performance with the best system in Marathi and Malayalam.

5. Conclusion

In this section, we mention some of the observations we made about our system. Our performance in ML category has been encouraging and this can be attributed to the effectiveness in designing a word to phone mapper and suitable backoff mechanism for missing units. There is a scope for improvement in the synthesis of SUS sentences. We haven't performed any data pruning or segmentation boundary adjustment for the current submission. Our future direction will be towards strengthening the existing framework with specific observations towards failure scenarios.

6. References

[1] A. Black and K. Tokuda, "The blizzard challenge 2005: Evaluating corpus-based speech synthesis on common databases," in *Proceedings of Interspeech*, 2005.

[2] C. L. Bennett and A. W. Black, "The blizzard challenge 2006," in *Proc. Blizzard Challenge*, 2006.

[3] M. Fraser and S. King, "The blizzard challenge 2007," *Proc. BLZ3-2007 (in Proc. SSW6)*, 2007.

[4] S. King, R. A. Clark, C. Mayo, and V. Karaiskos, "The blizzard challenge 2008," 2008.

[5] A. W. Black, S. King, and K. Tokuda, "The blizzard challenge 2009," 2009.

[6] S. King and V. Karaiskos, "The blizzard challenge 2012," 2012.

[7] K. Prahallad, A. Vadapalli, N. Elluru, G. Mantena, B. Pulugundla, P. Bhaskararao, H. Murthy, S. King, V. Karaiskos, and A. Black, "The blizzard challenge 2013–indian language task," in *Blizzard Challenge Workshop 2013*, 2013.

[8] R. A. Clark, K. Richmond, and S. King, "Festival 2–build your own general purpose unit selection speech synthesiser," 2004.

[9] T. Dutoit, V. Pagel, N. Pierret, F. Bataille, and O. Van der Vrecken, "The mbrola project: Towards a set of high quality speech synthesizers free of use for non commercial purposes," in *Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on*, vol. 3. IEEE, 1996, pp. 1393–1396.

[10] R. A. Clark, K. Richmond, and S. King, "Multisyn: Open-domain unit selection for the festival speech synthesis system," *Speech Communication*, vol. 49, no. 4, pp. 317–330, 2007.

[11] P. Tsiakoulis, S. Karabetsos, A. Chalamandaris, and S. Raptis, "An overview of the ilsp unit selection text-to-speech synthesis system," in *Artificial Intelligence: Methods and Applications*. Springer, 2014, pp. 370–383.

[12] A. J. Hunt and A. W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*, vol. 1. IEEE, 1996, pp. 373–376.

[13] S. P. Kishore and A. W. Black, "Unit size in unit selection speech synthesis." in *INTERSPEECH*, 2003.

Figure 2: Similarity and Naturalness boxplots of RD, SUS and ML for Bengali

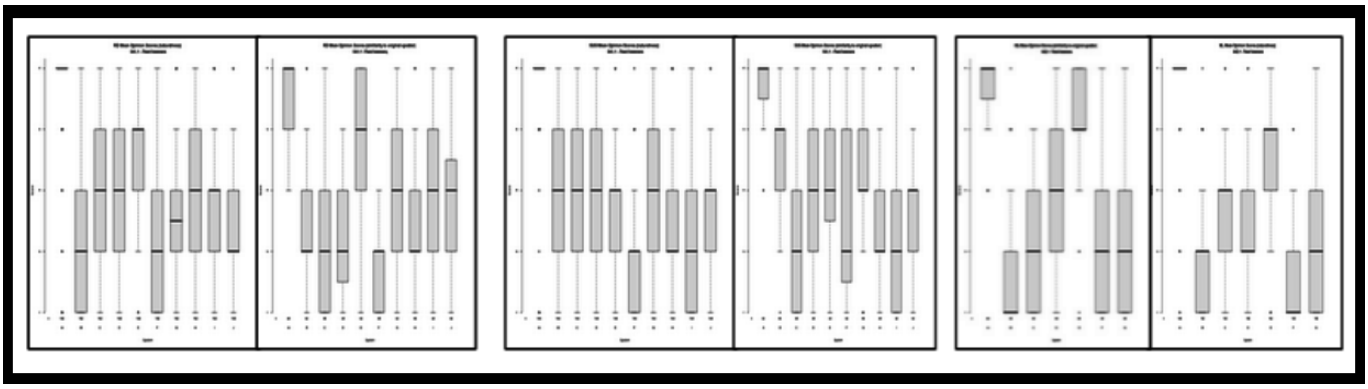


Figure 3: Similarity and Naturalness boxplots of RD, SUS and ML for Hindi

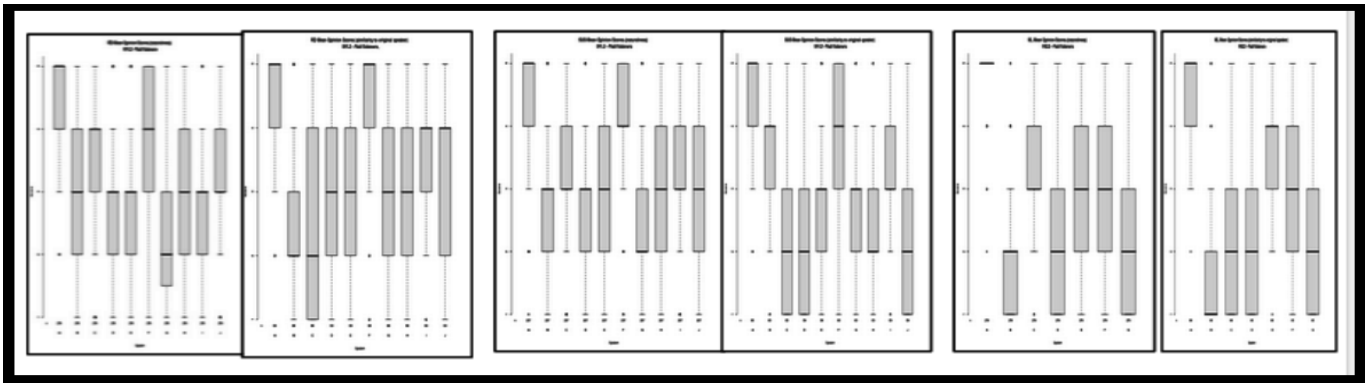


Figure 4: Similarity and Naturalness boxplots of RD, SUS and ML for Malayalam

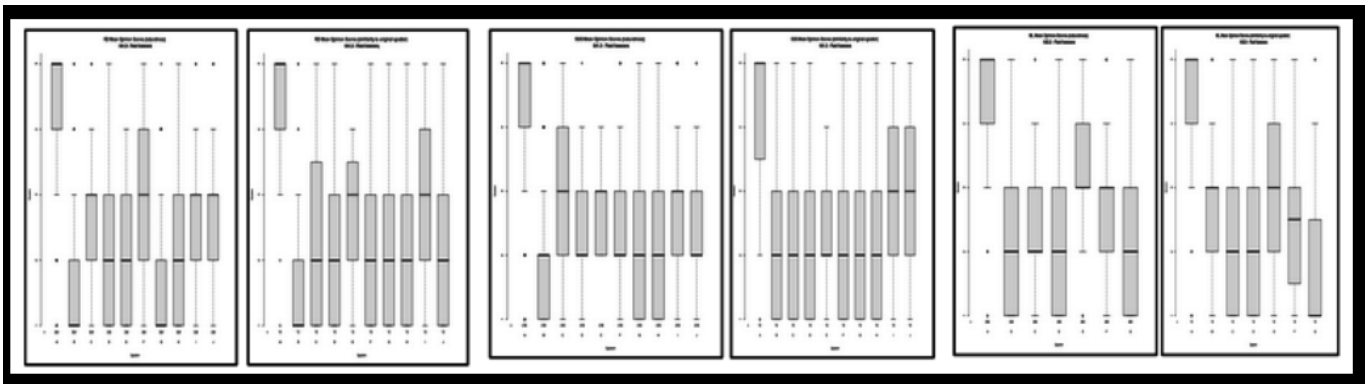


Figure 5: Similarity and Naturalness boxplots of RD, SUS and ML for Marathi

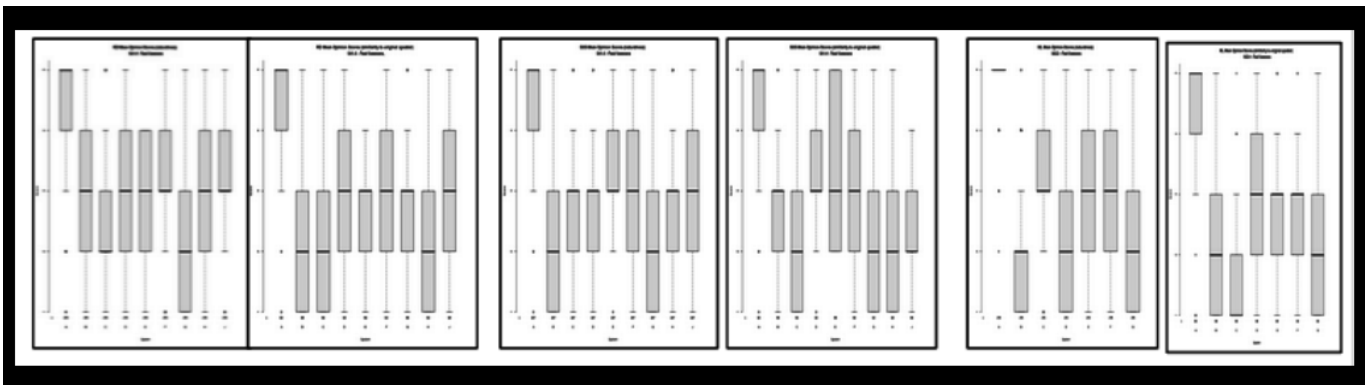


Figure 6: Similarity and Naturalness boxplots of RD, SUS and ML for Tamil

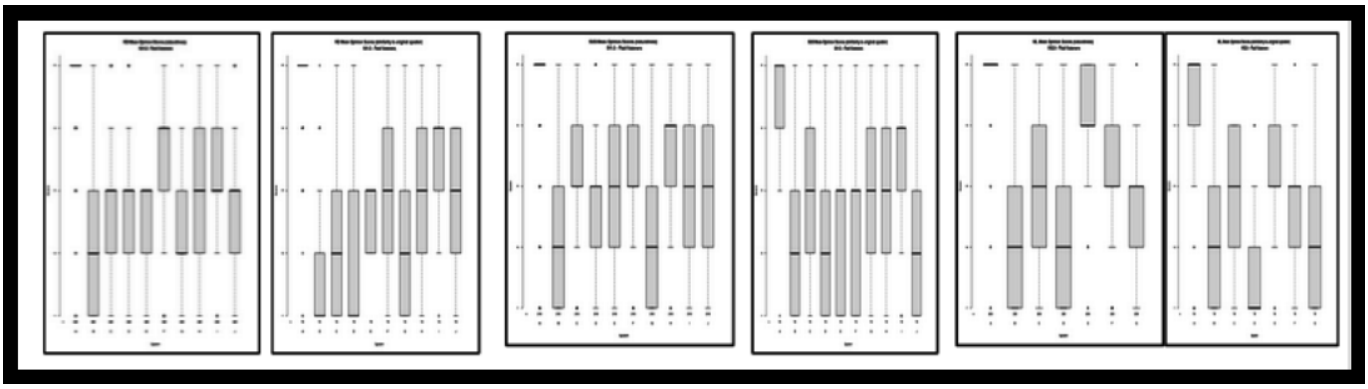


Figure 7: Similarity and Naturalness boxplots of RD, SUS and ML for Telugu

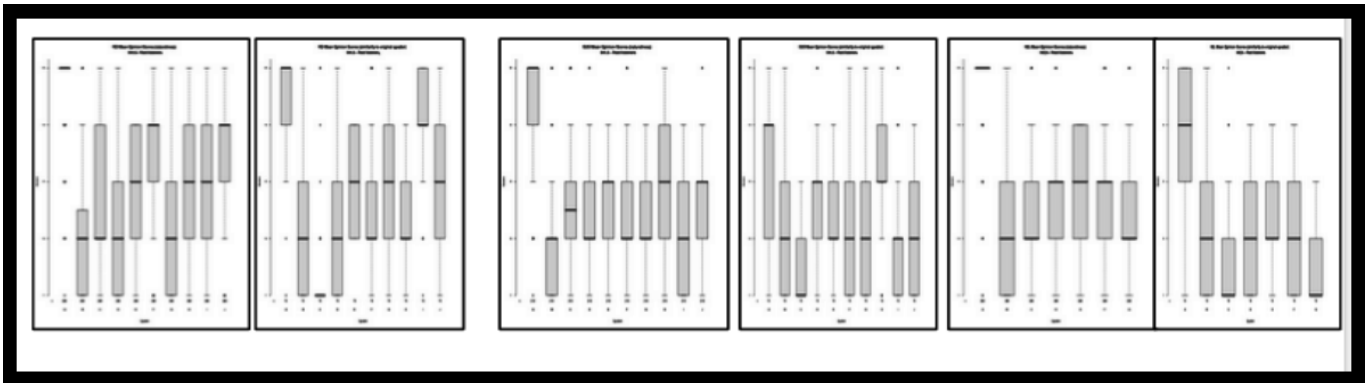
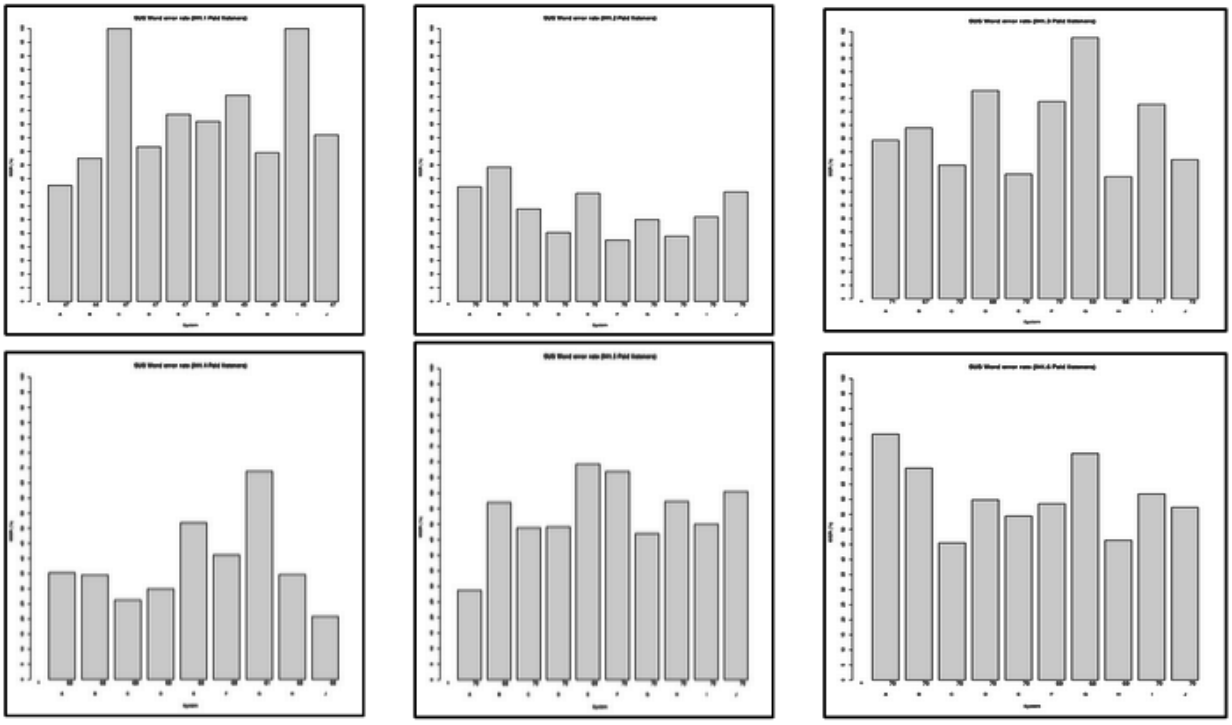


Figure 8: Word Error Rate boxplots for the languages



[14] K. Prahallad, A. R. Toth, and A. W. Black, "Automatic building of synthetic voices from large multi-paragraph speech databases." in *INTERSPEECH*, 2007, pp. 2901–2904.

[15] S. Kishore, R. Kumar, and R. Sangal, "A data driven synthesis approach for indian languages using syllable as basic unit," in *Proceedings of Intl. Conf. on NLP (ICON)*, 2002, pp. 311–316.

[16] H. A. Murthy, "Methods for improving the quality of syllable based speech synthesis," 2008.

[17] M. Vinodh, A. Bellur, K. Narayan, D. M. Thakare, A. Susan, N. Suthakar, H. Murthy *et al.*, "Using polysyllabic units for text to speech synthesis in indian languages," in *Communications (NCC), 2010 National Conference on*. IEEE, 2010, pp. 1–5.

[18] K. S. Rao and B. Yegnanarayana, "Modeling durations of syllables using neural networks," *Computer Speech & Language*, vol. 21, no. 2, pp. 282–295, 2007.

[19] A. Bellur, K. B. Narayan, K. Raghava Krishnan, H. Murthy *et al.*, "Prosody modeling for syllable-based concatenative speech synthesis of hindi and tamil," in *Communications (NCC), 2011 National Conference on*. IEEE, 2011, pp. 1–5.

[20] H. Kumar, J. Ashwini, B. Rajaramand, and A. Ramakrishnan, "Mile tts for tamil and kannada for blizzard challenge 2013," in *Blizzard Challenge Workshop 2013*, 2013.

[21] V. R. Lakkavalli, P. Arulmozhi, and A. Ramakrishnan, "Continuity metric for unit selection based text-to-speech synthesis," in *INTERSPEECH*, 2010.

[22] S. K. H. Rajaram, BSR and A. Ramakrishnan, "Mile tts for tamil for blizzard challenge 2014," in *Signal Processing and Communications (SPCOM), 2010 International Conference on*. IEEE, 2010, pp. 1–5.

[23] M. Schröder, A. Hunecke, and S. Krstulovic, "Openmary–open source unit selection as the basis for research on expressive synthesis," in *Proc. Blizzard Challenge*, vol. 6, 2006.

[24] V. Peddinti and K. Prahallad, "Significance of vowel epenthesis in telugu text-to-speech synthesis," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 5348–5351.

[25] N. K. Elluru, A. Vadapalli, R. Elluru, H. Murthy, and K. Prahallad, "Is word-to-phone mapping better than phone-phone mapping for handling english words?" in *ACL (2)*, 2013, pp. 196–200.

[26] A. W. Black, K. Lenzo, and V. Pagel, "Issues in building general letter to sound rules," 1998.

[27] W. Verhelst and M. Roelands, "An overlap-add technique based on waveform similarity (wsola) for high quality time-scale modification of speech," in *Acoustics, Speech, and Signal Processing, 1993. ICASSP-93., 1993 IEEE International Conference on*, vol. 2. IEEE, 1993, pp. 554–557.

[28] T. Hirai and S. Tenpaku, "Using 5 ms segments in concatenative speech synthesis," in *Fifth ISCA Workshop on Speech Synthesis*, 2004.