Overview of NITECH HMM-based text-to-speech system for Blizzard Challenge 2014

Kei Sawada[1], *Shinji Takaki*[1,2], *Kei Hashimoto*[1], *Keiichiro Oura*[1], *and Keiichi Tokuda*[1]

[1]Department of Scientific and Engineering Simulation,
Nagoya Institute of Technology, Nagoya, JAPAN
[2]Digital Content and Media Sciences Research Division,
National Institute of Informatics, Tokyo, JAPAN

{swdkei, k-prr44, bonanza, uratec, tokuda}@sp.nitech.ac.jp

Abstract

This paper describes a hidden Markov model based text-to-speech (TTS) system developed at the Nagoya Institute of Technology (NITECH) for Blizzard Challenge 2014. The tasks of Blizzard Challenge 2014 are speech synthesis of six Indian languages and multilingual speech synthesis, i.e., Indian language and English. Only Indian language speech data and text are provided as training data. We focused on constructing a TTS system without the phoneme information and phoneset of the target Indian languages. The proposed method is able to construct sentences written in a language with a space between words. The results of a large-scale subjectivity evaluation are discussed.

Index Terms: speech synthesis, hidden Markov model, grapheme-to-phoneme converter, multilingual speech synthesis, Blizzard Challenge

1. Introduction

In recent years, a number of studies have been conducted on text-to-speech (TTS) systems. TTS systems have been used widely in various applications, such as in-car navigation systems, mobile phone applications, and spoken dialogue systems. The demand for TTS systems of various speaking styles, e.g., emotion, and languages has increased in diverse fields.

Typical TTS systems have two main components, text analysis and speech waveform generation. In speech waveform generation, approaches based on unit-selection [1], hidden Markov models (HMMs) [2], deep neural network [3], etc. have been proposed. Since HMM-based speech synthesis has been actively researched in recent years, the synthetic speech quality of this method improved greatly. In HMM-based speech synthesis, the spectrum, excitation, and duration of speech are simultaneously modeled by using HMMs, and speech parameter sequences are generated from the HMMs themselves [4]. Compared with other synthesis methods, this method has several advantages. First, under its statistical training framework, it can learn the statistical properties of speakers, speaking styles [5], emotions [6], etc. from a speech corpus. Second, many techniques developed for HMM-based speech recognition can be applied to speech synthesis [7, 8]. Third, the voice characteristics of synthesized speech can be easily controlled by modifying the acoustic statistics of HMMs [9, 10].

The Blizzard Challenge was started in order to better understand and compare research techniques in building corpus-based speech synthesizers with the same data in 2005 [11, 12]. The Blizzard Challenge so far has provided English, Mandarin, audiobooks, etc. as a database. The tasks of Blizzard Challenge 2014 are speech synthesis of six Indian language (Assamese, Gujarati, Hindi, Rajasthani, Tamil, and Telugu) and multilingual speech synthesis, i.e., Indian language and English [13]. The provided databases [14] consist of Indian language speech data and text. That is, the databases do not included the phoneme information and phoneset of the target Indian languages. In typical HMM-based speech synthesis, HMMs are modeled at the phoneme-level. For this reason, phoneme information is required in order to train phoneme-level HMMs. Under normal circumstances, to define a phoneset fully requires special knowledge of the target language. Furthermore, labeling of speech data demands high cost. Therefore, obtaining phoneme information is difficult or impossible for the someone not familiar with the target language. TTS system building methods for target languages without an orthography have been proposed [15]. Nevertheless, a TTS system building method for when both the speech data and text of a target language exist has hardly investigated.

In this paper, we focus on constructing a TTS system without the phoneme information and phoneset of a target language. The problem with this situation is that label sequences of the target language speech and the lexicon of the target language do not exist. To obtain label sequences, speech recognition of target language speech is carried out by using the speech recognizer of another language, e.g., English. Consequently, the phoneset is the same as the recognizer one. Since a lexicon does not exist, label sequences of input text are generated by using a joint multigram grapheme-to-phoneme (G2P) converter [16].

In our method, first, a speech recognizer is built by using an English database. Then, the target Indian language speech is recognized by the English speech recognizer. Finally, a G2P converter and HMM-based speech synthesis system is built by using the recognition results. With these processes, it is possible to construct a TTS system without the phoneme information phoneset of the target language. This method is able to construct sentences written with in language with a space between words.

The rest of this paper is organized as follows. In Section 2, we explains briefly the rules of Blizzard Challenge 2014. In Section 3, we describes our speech synthesis system. Subjective listening test results are presented in Section 4. Concluding remarks and future work are presented in the final section.

2. Blizzard Challenge 2014 Rules

This year's challenge is the construction a TTS system for six Indian languages (Assamese, Gujarati, Hindi, Rajasthani, Tamil, and Telugu) [13, 14]. This challenge included two tasks: one for a Hub task (IH1) and one for a Spoke task (IH2) on Indian language.

The Hub task (IH1) was to build one voice in each Indian language from the provided speech data and the corresponding text in the UTF-8 format. About 2 hours of speech data, sampling at 16kHz, in each of the six Indian languages are provided as training data. The provided databases do not include the phoneme information and phoneset.

The Spoke task (IH2) was to build a multilingual speech synthesis, i.e., Indian language and English. The training data for this task was the same as for the Hub task. Training data do not contain any English words at all. The given sample input text to be synthesized for the Spoke task was as follows,

---

Sample input text for Spoke task (Hindi and English)

उन्हें 10 दिन तक rehab करना होगा और उसके बाद उनका fitness test लिया जाएगा

Sentences in the Indian are written with a space between words. The canonical shapes of the Indian language syllables are V, CV, CCV, and CCCV and thus have a generalized form of C*V, where C stands for a consonant and V for a vowel [17].

3. System Overview

Figure 1 shows an overview of the NITECH TTS system for Blizzard Challenge 2014. The TTS system for a target language (Indian) is made from an English database, speech data, and text of the target language. This system is constructed with a speech recognizer, word aligner, speech synthesizer, and grapheme-to-phoneme (G2P) converter. In the following sections, we describe the details of each component part.

3.1. Speech Recognizer

Since the label sequences of the target language speech do not exist, to obtain the label sequences, speech recognition of the target language speech is carried out by using an English speech recognizer. In our system, first, an initial speech recognizer is built by using an English database. To train the HTK [18] recognizer, we used the CMU pronunciation dictionary and the WSJ0, WSJ1, and TIMIT databases. The acoustic feature vector consists of 39 components comprised of 12-dimension mel-frequency cepstral coefficients (MFCCs) including the 0th order coefficient with the first and second order derivatives. The trained GMMs have 32 mixtures for silence and 16 mixtures for the others. This recipe is the same as that of the HTK Wall Street Journal Training Recipe [19]. The target language speech is then recognized with the initial speech recognizer obtained, in this way. For target language recognition, the network was designed so that it might be connected with every phoneme. In addition, the insertion penalty was set to minus 30.

The speech recognizer of a target language is trained by using the label sequences recognized by the initial speech recognizer. Furthermore, it can be expected that a high accuracy speech recognizer is re-trained by using the label sequences recognized by the speech recognizer. Our system estimates the speech recognizer of a target language twice.

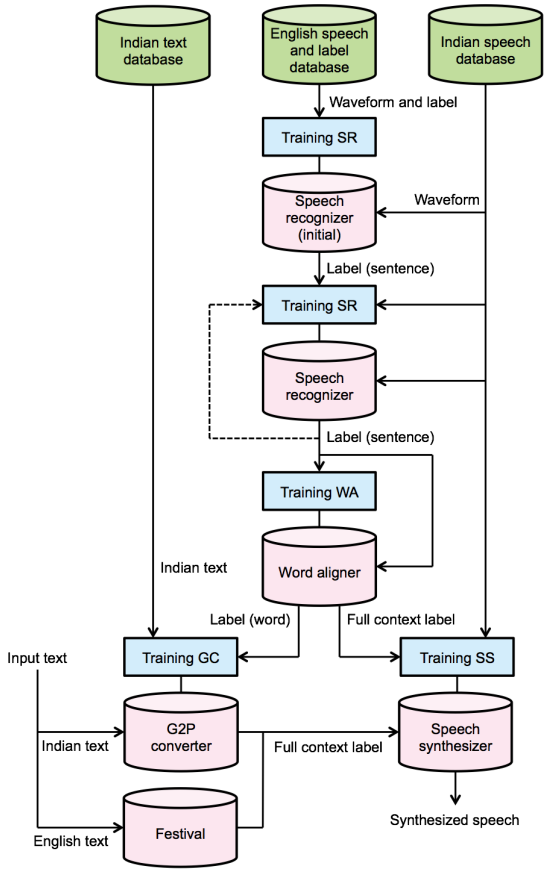

Figure 1: Overview of NITECH TTS system

3.2. Word Aligner

A label sequence $\boldsymbol{S}$ obtained by speech recognition does not include word breaking information, which is required for the full context labels of speech synthesis. Additionally, the text of speech synthesis is input word by word. Therefore, a word-level G2P converter is required.

The word aligner is estimated by using multigram models in order to obtain word breaking information [20]. A segmentation $\boldsymbol{q}$ of a string $\boldsymbol{S}$ into K sequences can be written as $\boldsymbol{q} = q_1, q_2, \ldots, q_k, \ldots, q_K$. The optimal alignment $\hat{\boldsymbol{q}}$ is estimated as follows:

$$\hat{\boldsymbol{q}} = \arg\max_{\boldsymbol{q} \in \boldsymbol{q}^*} \prod_{k=1}^{K} P(q_k). \quad (1)$$

Where, $\boldsymbol{q}^*$ denotes the set of all sequences. The parameters of the multigram models are estimated by using the expectation-maximization (EM) algorithm. Word alignment is obtained by applying the Viterbi algorithm. These steps are estimated by providing a constraint condition such that a pause of recognition results is word breaking.

3.3. Speech Synthesizer

Figure 2 overviews a HMM-based speech synthesis system. It consists of training and synthesis parts [21]. We used the HTS [22] for this system.

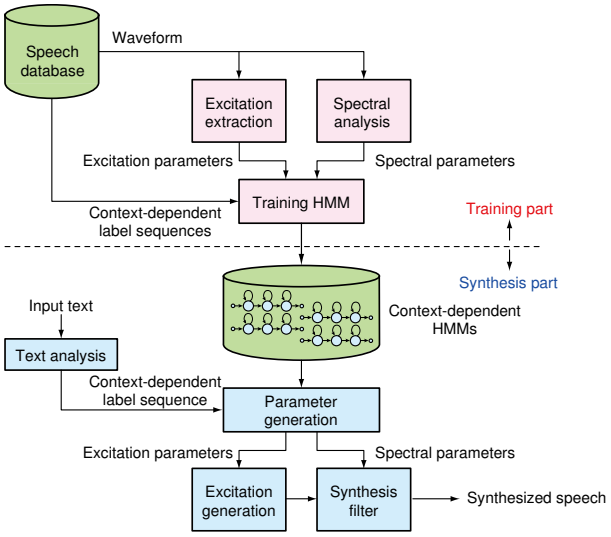

Figure 2: Overview of HMM-based speech synthesis system

The training part is similar to that used in speech recognition. The main difference is that both spectrum, e.g., mel-cepstral coefficients and their dynamic features, and excitation, e.g., log f0 and its dynamic features, parameters are extracted from a speech database and modeled by using HMMs. In our system, the hidden semi-Markov model (HSMM) based speech synthesis framework [7] was used. It makes it possible to estimate state output and duration probability distributions simultaneously. Although the spectrum part can be modeled by using a continuous HMM, the f0 part cannot be modeled by using a continuous or discrete HMM because the f0 observation sequence is composed of a one-dimensional continuous value and discrete symbol that represents unvoiced. To model such an observation sequence, multi-space probability distributions (MSDs) [23] are used for state-output distributions.

The synthesis part does the inverse operation of speech recognition. First, an arbitrarily given input text to be synthesized is converted to a context-dependent label sequence, and then, a sentence HMM is constructed by concatenating the context-dependent HMMs in accordance with the label sequence. Second, state durations of the sentence HMM are determined on the basis of the state-duration distributions. Third, the speech parameter generation algorithm generates sequences of spectral and excitation parameters that maximize their output probabilities under the constraints between static and dynamic features [24]. Finally, a speech waveform is synthesized directly from the generated spectral and excitation parameters by using a speech synthesis filter.

As a high-quality speech vocoding method, we use STRAIGHT, which is a proposed vocoder type algorithm [25]. It consists of three main components; f0 extraction, spectral and aperiodic analysis, and speech synthesis. Using the extracted f0, we use the STRAIGHT method to perform pitch-adaptive spectral analysis combined with a surface reconstruction method in the time-frequency domain to remove signal periodicity.

We applied a parameter generation algorithm considering the global variance (GV) of the generated parameters [26] for both the spectral and f0 parameter generation processes. To improve the estimation accuracy of GV models, we use the GV features calculated from only the speech region, excluding the silence and pause regions, and estimate the context-dependent GV models instead of a single global GV model. The context-dependent GV models are tied by using a the decision-tree based context clustering method in a similar way to acoustic model parameter tying.

In the HMM-based speech synthesis system, context-dependent models are generally used to capture a variety of contextual factors. In our system, the contexts are phoneme, syllable, word, phrase, and utterance. A syllable was defined as the C*V. The consonant or vowel of a phoneme was dependent on the phoneset of the CMU pronunciation dictionary.

Table 1: Number of listeners

Language	Paid listeners	All listeners
Assamese	106	115
Gujarati	50	50
Hindi	100	109
Rajasthani	101	110
Tamil	100	109
Telugu	100	106

3.4. Grapheme-to-phoneme Converter

To make label sequences of input text, a joint multigram G2P converter is constructed [16]. The optimal grapheme and phoneme pair alignment $\hat{\boldsymbol{w}}$ is estimated as follows:

$$\hat{\boldsymbol{w}} = \arg\max_{\boldsymbol{w}\in\boldsymbol{w}^*} \prod_{w\in\boldsymbol{w}} P(w). \quad (2)$$

Where, w is pair of a grapheme sequence and a phoneme sequence, $\boldsymbol{w}$ is a pair of possibly different lengths, and $\boldsymbol{w}^*$ denotes the set of all pair sequences. The joint multigram G2P converter is trained by using Sequitur G2P [27].

In Blizzard Challenge 2014, the input text of speech synthesis includes Indian language and English. As the input text is in the UTF-8 format, it is easy to identify the language from the Unicode point. The phoneme sequences of Indian language text are generated from the G2P converter, and the phoneme sequences of English text are generated from Festival [28]. Our system is able to synthesize multilingual speech, owing to using the phoneset of the CMU pronunciation dictionary.

A pause is not contained in a generated phoneme sequence. Therefore, a pause is inserted into a label sequence when any of the following conditions exist: 1) a comma, colon, and parenthesis are present; 2) before or after a word that is easy to enter pause before or after in a speech recognition result.

4. Blizzard Challenge 2014 Evaluation

4.1. Experimental Conditions

Large-scale subjective experiments were conducted by the Blizzard Challenge 2014 organization. Table 1 shows the number of listeners.

For HMM-based speech synthesis system training, speech data containing pruned speech that had a short amount of silence at the beginning and end and noisy speech was used.

Table 2: Amount of training data

Language	Number of sentences	Time
Assamese	1427	2h, 3m, 11s
Gujarati	450	2h, 1m, 33s
Hindi	875	2h, 0m, 31s
Rajasthani	1369	2h, 13m, 22s
Tamil	822	1h, 57m, 48s
Telugu	1470	3h, 6m, 32s

Table 2 indicates the amount of training data. Speech signals were sampled at a 16 kHz rate and windowed by using an f0-adaptive Gaussian window with a 5 ms shift. Feature vectors were comprised of 183-dimensions: 39-dimension STRAIGHT [25] mel-cepstral coefficients (plus the zero-th coefficient), log f0, 19-dimension mel-cepstral analysis aperiodicity measures, and their dynamic and acceleration coefficients. We used 5-state left-to-right context-dependent multi-stream MSD-HSMMs [7, 23] without skip transitions as acoustic models. Each state output probability distribution was composed of spectrum, f0, and aperiodicity streams. The spectrum and aperiodicity streams were modeled by using single multi-variate Gaussian distributions with diagonal covariance matrices. The f0 stream was modeled by using a multi-space probability distribution consisting of a Gaussian distribution for voiced frames and a discrete distribution for unvoiced frames. State durations were modeled by using a one-dimensional Gaussian distribution.

4.2. Experimental Results

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".

Table 3 indicates the score and standard deviation of evaluation results. In this table, RD, SUS, and ML correspond as follows.

- RD: read text
- SUS: semantically unpredictable sentences
- ML: multilingual sentences (Indian language and English)

In addition, system "A" and "**C**" correspond as follows.

- A: natural speech
- **C**: NITECH system

The WER results showed that our system "**C**" fell short of the other high-ranking systems. Errors in the speech recognition results and word alignment affected the accuracy of the G2P converter. Output errors of G2P converter lead to decrease of intelligibility. Since the WER of Rajasthani was the same as the other systems, adjusting the insertion penalty can be expected to improve the WER.

In the evaluation of the Hub task (IH1) similarity and naturalness, our system "**C**" was worse than the other high-ranking systems. For Rajasthani with an WER equivalent to other systems, the difference in MOS between our system "**C**" and the highest scoring system was about 0.6. In contrast, in the case of Gujarati, Hindi, Tamil, and Telugu with the lowest ranking WER, the difference in MOS between our system "**C**" and the highest scoring system was about 1.0. These results suggest that the low MOS was caused by low intelligibility.

The Spoke task (IH2) produced better results than did the Hub task (IH1). In particular, the evaluation of similarity produced good results. It can be presumed that synthesized speech of Indian language and English was similar because it uses the same acoustic models. These results show that the proposed method is effective for multilingual speech synthesis.

5. Conclusions

We described a HMM-based TTS system developed at the Nagoya Institute of Technology (NITECH) for Blizzard Challenge 2014. The system was built without the phoneme information and phoneset of the target language, and as evidenced from TTS systems of six Indian languages, the proposed method was effective. Improving the accuracy of the G2P converter and evaluation in other language will be future work.

6. Acknowledgements

The research leading to these results was partly funded by the Core Research for Evolutional Science and Technology (CREST) from the Japan Science and Technology Agency (JST).

7. References

[1] A. Hunt and A. W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," *Proceedings of ICASSP 1996*, vol. 1, pp. 373–376, 1996.

[2] K. Tokuda, Y. Nankaku, T. Toda, H. Zen, J. Yamagishi, and K. Oura, "Speech synthesis based on hidden Markov models," *Proceedings of IEEE*, vol. 101, no. 5, pp. 1234–1252, 2013.

[3] H. Zen, A. Senior, and M. Schuster, "Statistical parametric speech synthesis using deep neural networks," *Proceedings of ICASSP 2013*, pp. 7962–7966, 2013.

[4] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis," *Proceedings of Eurospeech 1999*, pp. 2347–2350, 1999.

[5] J. Yamagishi, K. Onishi, T. Masuko, and T. Kobayashi, "Acoustic modeling of speaking styles and emotional expressions in HMM-based speech synthesis," *IEICE Transactions on Information & Systems*, vol. E88-D, no. 3, pp. 502–509, 2005.

[6] R. Tsuzuki, H. Zen, K. Tokuda, T. Kitamura, M. Bulut, and S. Narayanan, "Constructing emotional speech synthesizers with limited speech database," *Proceedings of ICSLP*, vol. 2, pp. 1185–1188, 2004.

[7] H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Hidden semi-Markov model based speech synthesis," *Proceedings of ICSLP*, pp. 1185–1180, 2004.

[8] J. Yamagishi and T. Kobayashi, "Average-voice-based speech synthesis using HSMM-based speaker adaptation and adaptive training," *IEICE Transactions on Information & Systems*, vol. E90-D, no. 2, pp. 533–543, 2007.

Table 3: Evaluation results (All listeners)

Language	System	Hub task (IH1)					Spoke task (IH2)	
			Similarity		Naturalness		Similarity	Naturalness
		WER (SUS)	RD	SUS	RD	SUS	ML	
Assamese	A	0.51 ± 0.33	3.3 ± 1.5	3.6 ± 1.56	4.7 ± 0.61	4.8 ± 0.54	3.8 ± 1.5	4.9 ± 0.41
	B	0.86 ± 0.40	1.8 ± 1.1	1.5 ± 0.90	2.1 ± 1.04	1.8 ± 0.97	1.6 ± 1.0	1.9 ± 1.01
	C	**0.84 ± 0.15**	**2.8 ± 1.3**	**2.3 ± 1.24**	**3.3 ± 0.97**	**2.9 ± 1.09**	**2.5 ± 1.3**	**2.8 ± 1.04**
	D	0.69 ± 0.25	3.2 ± 1.3	2.3 ± 1.17	3.5 ± 0.99	2.6 ± 1.07	2.8 ± 1.2	2.7 ± 1.00
	E	0.76 ± 0.43	2.6 ± 1.3	2.3 ± 1.30	2.9 ± 0.99	2.3 ± 0.98	2.3 ± 1.1	2.2 ± 0.94
	F	0.67 ± 0.31	2.9 ± 1.3	2.5 ± 1.21	3.4 ± 1.04	2.6 ± 1.05	–	–
	G	0.74 ± 0.38	3.2 ± 1.4	2.6 ± 1.26	3.9 ± 0.95	3.0 ± 1.17	–	–
	I	0.69 ± 0.38	1.8 ± 1.0	1.6 ± 0.96	2.1 ± 1.03	1.8 ± 0.95	–	–
Gujarati	A	0.24 ± 0.20	2.9 ± 1.4	2.5 ± 1.4	4.7 ± 0.56	4.5 ± 0.86	3.7 ± 1.39	4.7 ± 0.51
	B	0.34 ± 0.18	3.0 ± 1.1	3.3 ± 1.1	2.6 ± 1.02	2.8 ± 1.03	2.7 ± 1.17	3.0 ± 1.02
	C	**0.59 ± 0.25**	**3.0 ± 1.2**	**3.0 ± 1.2**	**2.8 ± 1.01**	**2.8 ± 0.93**	**2.5 ± 1.11**	**2.6 ± 1.08**
	D	0.40 ± 0.26	2.7 ± 1.1	2.3 ± 1.1	2.8 ± 1.07	2.5 ± 1.07	2.3 ± 1.03	2.5 ± 1.02
	E	0.23 ± 0.16	3.5 ± 1.1	3.5 ± 1.0	3.5 ± 0.96	3.1 ± 1.09	3.5 ± 0.93	2.9 ± 1.01
	F	0.25 ± 0.26	2.8 ± 1.1	2.7 ± 1.2	3.4 ± 1.07	3.0 ± 1.04	–	–
	G	0.37 ± 0.26	3.7 ± 1.2	2.8 ± 1.3	3.8 ± 1.08	3.4 ± 0.94	–	–
	H	0.41 ± 0.22	3.5 ± 1.2	3.2 ± 1.4	2.5 ± 1.09	2.3 ± 1.11	–	–
	I	0.44 ± 0.36	2.8 ± 1.2	2.6 ± 1.1	2.7 ± 1.08	2.5 ± 1.05	–	–
Hindi	A	0.22 ± 0.22	4.3 ± 0.98	3.4 ± 1.41	4.5 ± 0.84	4.4 ± 1.0	3.7 ± 1.4	4.3 ± 1.16
	B	0.26 ± 0.24	2.4 ± 1.19	2.4 ± 1.17	2.0 ± 0.93	2.3 ± 1.0	1.9 ± 1.1	2.0 ± 0.93
	C	**0.40 ± 0.26**	**2.6 ± 1.15**	**2.6 ± 1.10**	**2.5 ± 1.05**	**2.4 ± 1.0**	**2.7 ± 1.1**	**2.6 ± 1.00**
	D	0.24 ± 0.21	4.0 ± 1.06	4.0 ± 0.98	3.6 ± 1.03	3.7 ± 1.1	3.3 ± 1.2	2.8 ± 1.08
	E	0.27 ± 0.23	3.2 ± 1.06	2.9 ± 1.12	3.1 ± 1.01	3.2 ± 1.0	2.5 ± 1.1	2.6 ± 1.06
	F	0.24 ± 0.21	3.4 ± 1.13	3.2 ± 1.13	3.2 ± 1.07	3.2 ± 1.1	1.9 ± 1.1	2.8 ± 1.45
	G	0.29 ± 0.23	3.4 ± 1.08	3.3 ± 1.23	3.7 ± 0.96	3.8 ± 1.0	–	–
	H	0.30 ± 0.24	2.1 ± 1.01	2.0 ± 1.03	2.2 ± 1.02	2.1 ± 1.1	–	–
	I	0.30 ± 0.21	3.1 ± 1.13	2.8 ± 1.18	2.2 ± 1.06	2.4 ± 1.0	–	–
	K	0.25 ± 0.20	2.4 ± 1.20	2.6 ± 1.25	3.4 ± 1.08	3.5 ± 1.1	2.4 ± 1.2	3.0 ± 1.07
Rajasthani	A	0.62 ± 0.32	4.4 ± 0.94	4.1 ± 1.14	4.2 ± 1.08	4.2 ± 1.04	4.3 ± 1.02	4.3 ± 1.09
	B	1.00 ± 0.19	2.6 ± 1.16	2.2 ± 1.12	2.3 ± 1.11	2.3 ± 1.17	2.2 ± 0.96	2.3 ± 1.07
	C	**0.67 ± 0.26**	**3.5 ± 1.04**	**3.4 ± 1.03**	**3.3 ± 0.99**	**3.3 ± 1.10**	**3.4 ± 1.01**	**3.3 ± 0.98**
	D	0.65 ± 0.26	3.6 ± 1.19	3.6 ± 1.08	3.7 ± 1.06	3.8 ± 1.01	3.4 ± 1.14	3.6 ± 1.00
	E	0.60 ± 0.26	3.6 ± 1.15	3.9 ± 1.01	3.7 ± 1.09	3.7 ± 1.02	3.4 ± 1.18	3.7 ± 1.00
	F	0.64 ± 0.21	4.0 ± 1.13	3.9 ± 1.19	3.9 ± 1.12	4.1 ± 1.05	3.1 ± 1.06	3.2 ± 1.06
	G	0.59 ± 0.22	3.7 ± 1.06	3.7 ± 1.09	3.9 ± 1.02	4.1 ± 0.98	–	–
	H	0.67 ± 0.24	3.1 ± 1.20	3.1 ± 1.20	3.1 ± 1.09	3.2 ± 1.17	–	–
	I	0.57 ± 0.29	3.3 ± 1.15	3.8 ± 0.93	3.2 ± 1.09	3.4 ± 1.11	–	–
Tamil	A	0.32 ± 0.33	4.0 ± 1.3	3.5 ± 1.4	4.6 ± 0.89	4.5 ± 0.92	4.0 ± 1.4	4.6 ± 0.81
	B	0.33 ± 0.27	2.0 ± 1.3	2.1 ± 1.3	2.3 ± 1.06	2.5 ± 1.15	2.2 ± 1.3	2.3 ± 1.10
	C	**0.64 ± 0.27**	**2.6 ± 1.5**	**2.6 ± 1.3**	**2.7 ± 1.14**	**2.9 ± 1.14**	**3.1 ± 1.3**	**2.6 ± 1.12**
	D	0.38 ± 0.31	3.0 ± 1.3	3.1 ± 1.4	3.2 ± 1.25	3.7 ± 1.11	3.1 ± 1.3	3.2 ± 1.13
	E	0.37 ± 0.30	2.7 ± 1.3	2.6 ± 1.4	2.9 ± 1.11	3.3 ± 1.12	2.6 ± 1.3	2.8 ± 1.11
	F	0.37 ± 0.28	2.7 ± 1.4	3.0 ± 1.4	3.4 ± 1.08	3.6 ± 1.11	–	–
	G	0.37 ± 0.32	3.8 ± 1.1	3.6 ± 1.2	3.6 ± 1.13	3.9 ± 1.09	–	–
	H	0.60 ± 0.29	3.2 ± 1.3	3.0 ± 1.4	2.7 ± 1.14	2.9 ± 1.18	–	–
	I	0.34 ± 0.31	1.8 ± 1.2	2.1 ± 1.3	2.6 ± 1.14	2.9 ± 1.19	–	–
	J	0.44 ± 0.32	3.1 ± 1.4	2.9 ± 1.3	2.6 ± 1.13	2.6 ± 1.13	2.7 ± 1.2	2.8 ± 1.10
Telugu	A	0.40 ± 0.25	4.5 ± 0.73	4.6 ± 0.69	4.9 ± 0.39	4.8 ± 0.47	4.7 ± 0.73	4.9 ± 0.36
	B	0.55 ± 0.29	1.7 ± 0.84	1.5 ± 0.76	2.0 ± 0.85	1.8 ± 0.81	1.6 ± 0.84	2.0 ± 0.80
	C	**0.77 ± 0.21**	**2.6 ± 1.07**	**2.1 ± 1.00**	**3.1 ± 0.98**	**2.5 ± 1.09**	**2.4 ± 1.06**	**2.5 ± 1.08**
	D	0.54 ± 0.25	2.5 ± 1.14	2.1 ± 0.91	3.5 ± 0.89	2.8 ± 0.96	3.0 ± 0.96	3.1 ± 0.99
	E	0.51 ± 0.26	2.3 ± 0.95	1.9 ± 0.94	3.1 ± 0.90	2.6 ± 0.94	2.6 ± 0.96	3.1 ± 0.86
	F	0.46 ± 0.25	3.3 ± 1.11	2.9 ± 1.12	4.0 ± 0.85	3.2 ± 1.01	1.9 ± 0.90	2.3 ± 0.90
	G	0.51 ± 0.27	3.9 ± 0.93	2.8 ± 1.10	4.2 ± 0.81	3.4 ± 1.00	–	–
	H	0.57 ± 0.39	1.4 ± 0.68	1.3 ± 0.53	1.9 ± 0.85	1.6 ± 0.72	–	–
	I	0.62 ± 0.22	2.9 ± 1.30	2.2 ± 1.20	2.3 ± 0.98	1.8 ± 0.89	–	–

[9] M. Tamura, T. Masuko, K. Tokuda, and T. Kobayashi, "Speaker adaptation for HMM-based speech synthesis system using MLLR," *Proceedings of ESCA/COCOSDA Third International Workshop on Speech Synthesis*, pp. 273–276, 1998.

[10] ——, "Adaptation of pitch and spectrum for HMM-based speech synthesis using MLLR," *Proceedings of ICASSP 2001*, pp. 805–808, 2001.

[11] A. W. Black and K. Tokuda, "The Blizzard Challenge – 2005: Evaluating corpus-based speech synthesis on common datasets," *Proceedings of Interspeech 2005*, pp. 77–80, 2005.

[12] Blizzard Challenge Website. [Online]. Available:, http://synsig.org/index.php/Blizzard_Challenge.

[13] Blizzard Challenge 2014. [Online]. Available:, http://www.synsig.org/index.php/Blizzard_Challenge_2014.

[14] H. A. Patil, T. B. Patel, N. J. Shah, H. B. Sailor, R. Krishnan, G. R. Kasthuri, T. Nagarajan, L. Christina, N. Kumar, V. Raghavendra, S. P. Kishore, S. R. M. Prasanna, N. Adiga, S. R. Singh, K. Anand, P. Kumar, B. C. Singh, S. L. Binil Kumar, T. G. Bhadran, T. Sajini, A. Saha, T. Basu, K. S. Rao, N. P. Narendra, A. K. Sao, R. Kumar, P. Talukdar, P. Acharyaa, S. Chandra, S. Lata, and H. A. Murthy, "A syllable-based framework for unit selection synthesis in 13 Indian languages," *Proceedings of O-COCOSDA/CASLRE*, pp. 1–8, 2013.

[15] S. Sitaram, S. Palkar, Y. Chen, A. Parlikar, and A. W. Black, "Bootstrapping text-to-speech for speech processing in languages without an orthography," *Proceedings of ICASSP 2013*, pp. 7992–7996, 2013.

[16] M. Bisani and H. Ney, "Joint-sequence models for grapheme-to-phoneme conversion," *Proceedings of Speech Communication*, vol. 50, no. 5, pp. 434–451, 2008.

[17] K. Prahallad, A. Vadapalli, N. Elluru, G. Mantena, B. Pulugundla, P. Bhaskararao, H. A. Murthy, S. King, V. Karaiskos, and A. W. Black, "The Blizzard Challenge 2013 - Indian language tasks," *Proceedings of Blizzard Challenge 2013 Workshop*, 2013.

[18] HTK. [Online]. Available:, http://htk.eng.cam.ac.uk/.

[19] K. Vertanen, "Baseline WSJ acoustic models for HTK and Sphinx: Training recipes and recognition experiments," *Proceedings of Cavendish Laboratory*, 2006.

[20] S. Deligne and F. Bimbot, "Language modeling by variable length sequences : Theoretical formulation and evaluation of multigrams," *Proceedings of ICASSP 1995*, pp. 169–172, 1995.

[21] S. Takaki, K. Sawada, K. Hashimoto, K. Oura, and K. Tokuda, "Overview of NITECH HMM-based speech synthesis system for Blizard Challenge 2013," *Proceedings of Blizzard Challenge 2013 Workshop*, 2013.

[22] HTS. [Online]. Available:, http://hts.sp.nitech.ac.jp/.

[23] K. Tokuda, T. Masuko, N. Miyazaki, and T. Kobayashi, "Multi-space probability distribution HMM," *IEICE Transactions on Information & Systems*, vol. E85-D, no. 3, pp. 455–464, 2002.

[24] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," *Proceedings of ICASSP 2000*, pp. 936–939, 2000.

[25] H. Kawahara, I. Masuda-Katsuse, and A. Cheveigne, "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, pp. 187–207, 1999.

[26] T. Toda and K. Tokuda, "Speech parameter generation algorithm considering global variance for HMM-based speech synthesis," *Proceedings of Interspeech 2005*, pp. 2801–2804, 2005.

[27] Sequitur G2P. [Online]. Available:, http://www-i6.informatik.rwth-aachen.de/web/Software/g2p.html.

[28] Festival. [Online]. Available:, http://www.festvox.org/festival/.