The NITECH HMM-based text-to-speech system for the Blizzard Challenge 2015

Kei Sawada, Kei Hashimoto, Keiichiro Oura, and Keiichi Tokuda

Department of Scientific and Engineering Simulation,
Nagoya Institute of Technology, Nagoya, JAPAN
{swdkei, bonanza, uratec, tokuda}@sp.nitech.ac.jp

Abstract

This paper describes a hidden Markov model (HMM)-based text-to-speech (TTS) system developed at the Nagoya Institute of Technology (NITECH) for the Blizzard Challenge 2015. The tasks of the Blizzard Challenge 2015 are speech synthesis for six Indian languages and multilingual involving one Indian language and English. In this challenge, only Indian language speech data and text are provided as training data. Therefore, pronunciation information, such as phoneset, phoneme sequences, and lexicon, for each language is needed to construct TTS systems. Standard methods for constructing a TTS system of a new language take a huge cost because it requires a special knowledge of the target language. In this paper, we focus on automatic construction of a TTS system without the special knowledge of the target language. The results of a large-scale subjectivity evaluation are discussed.

Index Terms: text-to-speech system, unknown-pronunciation language, multilingual speech synthesis, hidden Markov model, Blizzard Challenge

1. Introduction

In recent years, a number of studies for text-to-speech (TTS) systems have been conducted. Consequently, quality of synthetic speech has been improved and TTS systems have been widely used in various applications. For expanded use of applications, the demand for TTS systems of various languages has increased in diverse fields. It is considered that thousands of languages exist in the world [1]. However, in traditional methods, it is difficult to build TTS systems of any language. Therefore, one goal of the speech synthesis research is to establish a framework that can be applied to build TTS systems of any language.

Typical TTS systems have two main components: text analysis and speech waveform generation parts. In the text analysis part, pronunciation of an input text is estimated by using a lexicon which contain pronunciation (phoneme) information. In addition, some context information, e.g., part of speech and accent, is obtained in this part. Since this part is highly dependent on the target language, construction of it requires a huge cost for someone not familiar with the target language. In the speech waveform generation part, speech waveforms are generated from the pronunciation estimated by the text analysis part. Approaches based on unit-selection [2] and statistical parametric speech synthesis (SPSS), e.g., hidden Markov model (HMM)- [3] and deep neural network (DNN)- [4] based speech synthesis, have been proposed for the speech waveform generation part. Since the HMM-based speech synthesis has been actively researched in recent years, the synthetic speech quality of this method improved greatly. 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]. Fourth, supporting multiple languages can easily be accomplished because the only language-dependent element is the set of contextual factors to be used.

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]. This challenge so far has provided English, Mandarin, audiobooks, etc. as a database. The tasks of the Blizzard Challenge 2015 are speech synthesis for six Indian languages (Bengali, Hindi, Malayalam, Marathi, Tamil, and Telugu) and multilingual involving one Indian language and English [13]. The provided databases [14] consist of Indian language speech data and text. That is, pronunciation information, such as phoneset, phoneme sequences, and lexicon, for each language is needed to construct TTS systems.

Typical HMM-based TTS systems require phoneme information because acoustic features are generally modeled at the phoneme-level. Under normal circumstances, to define a phoneset fully requires a special knowledge of the target language. Even if a phoneset can be defined, labeling of target speech data demands a high cost. Therefore, obtaining some phoneme information is difficult or impossible for someone not familiar with the target language. To establish a low language-dependency framework to construct TTS systems from a database that consist of only speech data and text is important for the speech synthesis research. In this paper, we focus on automatic construction of a TTS system without a special knowledge of any unknown-pronunciation languages. The problem in this situation is that a phoneset, label sequences corresponding to target speech data, and a lexicon do not exist. To solve a phoneset and label sequences problem, speech recognition is carried out by using the speech recognizer of the other language, e.g., English. Label sequences of target speech data can be obtained by using speech recognition. Then, an HMM-based speech synthesizer of the target language is trained by using pairs of speech data and label sequences. To solve a lexicon problem, a joint multigram grapheme-to-phoneme converter is trained by using pairs of text and label sequences [15]. With these processes, it is possible to construct a TTS system without a special knowledge of the target language.

The rest of this paper is organized as follows. In Section 2,

the tasks of the Blizzard Challenge 2015 are briefly explained. Section 3 describes our TTS system. Subjective listening test results are presented in Section 4. Concluding remarks and future work are presented in the final section.

2. Blizzard Challenge 2015 tasks

The Blizzard Challenge 2015 is the construction of TTS systems for six Indian languages (Bengali, Hindi, Malayalam, Marathi, Tamil, and Telugu) [13]. This challenge includes two tasks: one for a Hub task and one for a Spoke task on Indian language.

The Hub task is to build TTS systems in each Indian language from the provided speech data and the corresponding text in UTF-8 format. About four or two hours of speech data, sampled at 16kHz, in each of the six Indian languages are provided as training data.

The Spoke task is to build multilingual (polyglot) TTS systems, i.e., one Indian language and English. Training data for this task is the same as for the Hub task, i.e., the training data do not contain any English words at all. The example input text to be synthesized for the Spoke task is as follows:

Example input text for the Spoke task (Hindi and English)

Stanford वैज्ञानिकों द्वारा विकसित नयी aluminium battery, केवल एक minute में cellphone को charge कर सकती है.

---

3. System overview

3.1. Construction of HMM-based text-to-speech system

In this paper, we propose a TTS system building method using a target language database which consists of speech data and text corresponding to speech data. In typical HMM-based TTS systems, acoustic features are modeled at the phoneme-level. Thus, the following steps are needed in order to construct a TTS system of a new language.

Step 1 Definition of a phoneset.

Step 2 Construction of a lexicon or grapheme-to-phoneme converter for the text analysis part.

Step 3 Definition of contextual factors to be considered in acoustic modeling.

Step 4 Preparation of label sequences corresponding to speech data.

These steps take a huge cost for someone not familiar with the target language because it requires a special knowledge of the target language. Therefore, to establish a low language-dependency framework of a TTS system construction from a database consist of speech data and text is important for the speech synthesis research. In this paper, we investigate a framework to automatic construction of TTS systems for any unknown-pronunciation languages.

3.2. Construction of HMM-based text-to-speech system in unknown-pronunciation language

It is difficult to define phonesets for an unknown-pronunciation language. Furthermore, it is hard to obtain a phoneme sequence corresponding to the speech data. To solve these problems, our system automatically obtain phoneme sequences by using a speech recognizer for a different language, e.g., English, from

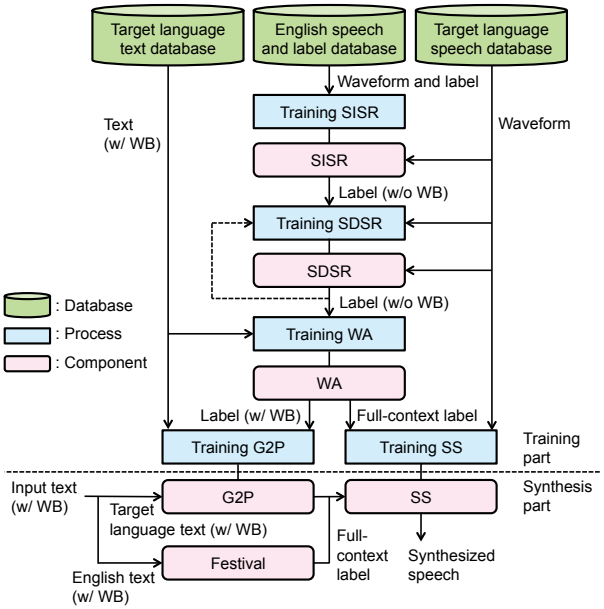

Figure 1: Overview of the NITECH TTS system.

the target language. Thereby, the phoneset of the different language speech recognizer is used as the phoneset of the target language. Although the phoneset is different from the correct phoneset of target language, similar phonemes are assigned to speech data in this approach. Figure 1 shows an overview of the NITECH TTS system. This system is constructed with a speech recognizer (SR), word aligner (WA), grapheme-to-phoneme converter (G2P), and speech synthesizer (SS). We call the SR, WA, and G2P as front-end components and call the SS as a back-end component. The details of each component are described in the following sections.

3.2.1. Speech recognizer (SR)

In the case of HMM-based SS, phoneme sequences corresponding to the speech data are necessary for acoustic modeling at the phoneme-level. To obtain label sequences, speech recognition is carried out by using a speaker-independent SR (SISR) of the other language. For the target language recognition, the phone network is designed so that it might be connected with every phoneme (triphone recognizer). Because of this process, the phoneset of SISR is used as the phoneset of the target language.

Since accuracy of phoneme sequences affect the latter components, i.e., the WA, G2P, and SS, it is important to estimate the high accuracy phoneme sequences. To estimate more accurate phoneme sequences, a speaker-dependent SR (SDSR) is constructed from initial phoneme sequences obtained by the SISR. Furthermore, estimation of phoneme sequences and training of the SDSR are iterated in order to improve the accuracy of phoneme sequences.

Modeling of phoneme durations is important for an SS. Thus, it is expected that estimation of phoneme sequences taking account of phoneme duration can obtain phoneme sequences suitable for the SS. However, a standard SR is difficult to consider a phoneme duration. Therefore, in our system, a phoneme sequence is selected using an alignment likelihood of a hidden semi-Markov model (HSMM) that have explicit duration distributions. Among the N-best hypotheses of the

speech recognition result, the phoneme sequence with the highest alignment likelihood is selected as the phoneme sequence corresponding to the speech data.

3.2.2. *Word aligner (WA)*

Since many languages are written with space between words, a word-level G2P is suitable for the text analysis part. Furthermore, word boundary (WB) information is useful for contextual factors of an SS. However, a label sequence obtained by the speech recognition does not include a WB. Therefore, a WA is constructed for estimation of WB. In our system, the WA is constructed by using a joint multigram models [15]. The optimal grapheme and phoneme pair $\hat{\boldsymbol{w}}$ is estimated as follows:

$$\hat{\boldsymbol{w}} = \arg\max_{\boldsymbol{w} \in \boldsymbol{w}^*} \prod_{w \in \boldsymbol{w}} P(w). \quad (1)$$

Where, w is a pair of a grapheme and a phoneme, $\boldsymbol{w}$ is a pair of possibly different lengths, and $\boldsymbol{w}^*$ denotes the set of all pair sequences. The parameters of the joint multigram models are estimated by using the expectation-maximization (EM) algorithm. The WA is estimated by providing a constraint condition such that a pause of recognition results must be WB. Then, a word alignment is obtained by applying the Viterbi algorithm.

3.2.3. *Grapheme-to-phoneme converter (G2P)*

To synthesize an arbitrary text, an input text needs to be converted into a phoneme sequence. However, in the unknown-pronunciation language, it is difficult to construct a lexicon for converting the input text into phonemes. To overcome this problem, a G2P based on a joint multigram model [15] is introduced instead of a lexicon. The joint multigram G2P is trained by using the Sequitur G2P [16].

Since training data of the G2P does not contain any pauses, pauses are not contained in generated phoneme sequences by the G2P. 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 in a speech recognition result.

In the Blizzard Challenge 2015, the input text to be synthesized includes Indian language and English. In our system, since initial phoneme sequences are obtained from English SISR, the phoneset of English is used a phoneset of Indian language. Therefore, our system is able to synthesize Indian language and English speech by converting the text into the phoneme sequence. The phoneme sequences of Indian language text are generated from the G2P, and the phoneme sequences of English text are generated from the Festival [17].

3.2.4. *Speech synthesizer (SS)*

In the case of HMM-based SS, a context-dependent models are generally used to capture a variety of contextual factors. A full-context label corresponding to the speech data in the target language is created from a phoneme sequence with WB information obtained by the SR and WA. The contexts of phoneme, syllable, word, phrase, and utterance are used in full-context labels. A syllable is defined as the C^*V, where C is a consonant, V is a vowel, and C^* indicates there may be none or more consonant. The consonant or vowel of a phoneme is dependent on the phoneset of the language used in SISR. The SS can be built in the same procedure as the standard procedure by using speech data and corresponding full-context labels. Figure 2 overviews

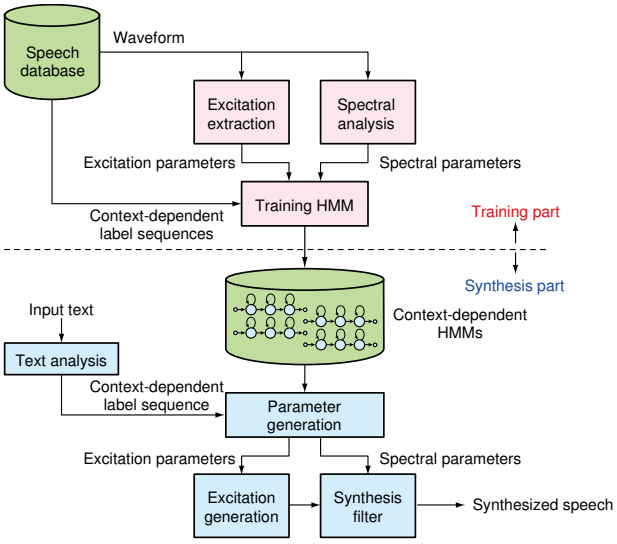

Figure 2: Overview of HMM-based SS.

an HMM-based speech synthesis system. It consists of training and synthesis parts. We used the HTS [18] for this component.

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 F_0 and its dynamic features, parameters are extracted from a speech database and modeled by using HMMs [19]. In our system, the HSMM-based speech synthesis framework [7] is used. It makes possible to estimate state output and duration probability distributions simultaneously. Although spectral parameters can be modeled by using a continuous HMM, F_0 cannot be modeled by using a continuous or discrete HMM because the F_0 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) [20] 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 by using G2P, 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 [21]. 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 vocoder type algorithm [22]. It consists of three main components: F_0 extraction, spectral and aperiodic analysis, and speech synthesis. Using the extracted F_0, 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 [23] for both the spectral and F_0 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 decision-tree based context clustering method in a similar way to acoustic model parameter tying.

4. Blizzard Challenge 2015 evaluation

4.1. Experimental conditions of speech recognizer

The target language was six Indian languages (Bengali, Hindi, Malayalam, Marathi, Tamil, and Telugu). Indian language databases were provided from the Blizzard Challenge 2015 organization [14]. To train the English SISR, we used the CMU pronunciation dictionary and the WSJ0, WSJ1, and TIMIT databases. Speech signals were sampled at a rate of 16 kHz and windowed by a 25-ms hamming window with a 10-ms shift. 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. A 3-state left-to-right HMM without skip transitions was used. The trained GMMs had 32 mixtures for silence and 16 mixtures for the others. We used the HTK [24] for this component. This recipe is the same as that of the HTK Wall Street Journal Training Recipe [25]. A phoneme sequence was selected using an alignment likelihood of the HSMM from 50-best hypothesis of speech recognition results. Table 1 indicates the insertion penalty and the number of iterations of estimation of phoneme sequences and training of the SDSR.

4.2. Experimental conditions of speech synthesizer

Speech signals were sampled at a 16 kHz rate and windowed by using an F_0-adaptive Gaussian window with a 5-ms shift. Feature vectors were comprised of 183-dimensions: 39-dimension STRAIGHT [22] mel-cepstral coefficients include the 0th coefficient, log F_0, 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, 20] without skip transitions as acoustic models. Each state output probability distribution was composed of spectrum, F_0, and aperiodicity streams. The spectrum and aperiodicity streams were modeled by using single multi-variate Gaussian distributions with diagonal covariance matrices. The F_0 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 1-dimensional Gaussian distribution. Table 2 indicates the amount of training data.

4.3. Experimental conditions of listening test

Large-scale subjective experiments were conducted by the Blizzard Challenge 2015 organization. Table 3 shows the number of paid listeners. To evaluate intelligibility, the subjects were asked to transcribe semantically unpredictable sentences by typing in the sentence they heard; the average word error rate (WER) were calculated from these transcripts. Furthermore, to evaluate the speaker 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 1: Insertion penalty and number of iterations

Language	Insertion penalty	# of iterations
Bengali	-20	3
Hindi	-40	2
Malayalam	-40	3
Marathi	-40	3
Tamil	-20	3
Telugu	-10	2

Table 2: Amount of training data

Language	# of sentences	Time
Bengali	1284	1h, 58m, 27s
Hindi	1690	3h, 57m, 59s
Malayalam	1269	1h, 58m, 9s
Marathi	1178	2h, 6m, 7s
Tamil	1440	4h, 9m, 28s
Telugu	2461	4h, 11m, 34s

Table 3: Number of paid listeners

Language	# of listeners
Bengali	48
Hindi	69
Malayalam	72
Marathi	69
Tamil	70
Telugu	70

4.4. Experimental results

Table 4 indicates the score and standard deviation of evaluation results. In this table, RD, SUS, and ML correspond as follows.

- RD: read text (Hub task)
- SUS: semantically unpredictable sentences (Hub task)
- ML: multilingual sentences, i.e., one Indian language and English (Spoke task)

In addition, system A, B, and $\boldsymbol{G}$ correspond as follows.

- A: natural speech
- B: baseline system
- $\boldsymbol{G}$: NITECH system

In our system, pronunciation errors can occur due to errors in the front-end component. It is considered that pronunciation errors lead the high WER and low MOS. It can actually be seen in Table 4 that our system $\boldsymbol{G}$ achieved equivalent result compared with the other systems in Hindi and Tamil. Especially in

Table 4: Evaluation results

Language	System	WER (%)	MOS of speaker similarity			MOS of naturalness		
		SUS	RD	SUS	ML	RD	SUS	ML
Bengali	A	43	4.5 ± 0.68	4.7 ± 0.54	4.6 ± 0.67	4.7 ± 0.86	4.6 ± 0.84	4.7 ± 0.78
	B	52	2.5 ± 1.19	3.7 ± 0.91	1.8 ± 1.12	2.2 ± 1.02	2.7 ± 1.26	1.8 ± 0.92
	C	100	2.2 ± 1.02	2.2 ± 1.15	2.1 ± 0.87	2.9 ± 1.06	3.1 ± 1.10	2.6 ± 1.15
	D	57	2.2 ± 0.90	3.0 ± 1.20	2.8 ± 0.99	3.0 ± 1.11	3.0 ± 1.04	2.6 ± 1.00
	E	69	3.7 ± 1.22	3.3 ± 1.09	4.1 ± 0.95	3.4 ± 1.07	2.7 ± 0.94	3.8 ± 0.95
	F	66	1.7 ± 0.78	2.6 ± 1.32	2.1 ± 1.03	2.0 ± 1.10	1.8 ± 0.88	1.6 ± 0.85
	$\boldsymbol{G}$	**76**	$\mathbf{3.1 \pm 1.21}$	$\mathbf{3.1 \pm 0.93}$	$\mathbf{2.2 \pm 1.21}$	$\mathbf{2.5 \pm 1.10}$	$\mathbf{2.8 \pm 1.00}$	$\mathbf{2.2 \pm 0.99}$
	H	55	2.4 ± 1.05	2.5 ± 1.18	–	2.6 ± 1.11	2.6 ± 1.06	–
	I	100	2.9 ± 1.08	2.3 ± 1.14	–	2.7 ± 1.08	2.1 ± 0.93	–
	J	61	2.7 ± 1.07	2.8 ± 1.04	–	2.6 ± 1.08	2.5 ± 0.93	–
Hindi	A	42	4.5 ± 0.82	4.4 ± 0.79	4.5 ± 0.72	4.7 ± 0.59	4.4 ± 0.82	4.7 ± 0.52
	B	49	2.6 ± 1.11	3.5 ± 1.01	1.8 ± 1.15	3.2 ± 1.20	2.6 ± 1.08	1.8 ± 0.94
	C	34	2.4 ± 1.27	2.0 ± 1.06	2.2 ± 1.04	3.5 ± 1.03	3.3 ± 1.10	3.2 ± 1.00
	D	25	2.7 ± 1.19	2.2 ± 1.16	2.0 ± 1.12	2.8 ± 1.09	2.7 ± 1.08	2.3 ± 1.17
	E	40	2.9 ± 1.17	2.8 ± 1.06	3.5 ± 1.07	2.6 ± 1.18	3.0 ± 1.17	3.2 ± 1.07
	F	23	4.3 ± 0.97	3.9 ± 1.14	3.1 ± 1.20	3.9 ± 0.92	3.9 ± 0.92	2.9 ± 1.25
	$\boldsymbol{G}$	**30**	$\mathbf{2.8 \pm 1.06}$	$\mathbf{2.9 \pm 1.06}$	$\mathbf{2.2 \pm 1.14}$	$\mathbf{2.3 \pm 1.01}$	$\mathbf{2.4 \pm 1.01}$	$\mathbf{2.0 \pm 0.93}$
	H	24	2.7 ± 1.33	2.6 ± 1.18	–	2.8 ± 1.11	3.0 ± 1.05	–
	I	31	3.5 ± 1.11	3.3 ± 1.10	–	2.8 ± 1.02	3.2 ± 1.09	–
	J	40	3.3 ± 1.17	2.2 ± 1.03	–	3.3 ± 0.97	3.1 ± 1.10	–
Malayalam	A	59	4.6 ± 0.85	4.2 ± 1.21	4.2 ± 1.2	4.3 ± 0.97	4.3 ± 1.09	4.4 ± 0.99
	B	64	1.8 ± 1.04	2.1 ± 1.20	2.6 ± 1.2	1.6 ± 0.88	1.9 ± 1.06	1.9 ± 0.83
	C	50	2.3 ± 1.30	2.1 ± 1.17	2.2 ± 1.3	2.6 ± 1.07	2.8 ± 1.00	2.4 ± 0.91
	D	78	2.1 ± 1.06	2.2 ± 1.19	2.1 ± 1.2	2.3 ± 1.09	2.4 ± 0.97	2.2 ± 1.06
	E	47	2.9 ± 0.80	2.6 ± 0.78	3.2 ± 1.0	2.3 ± 1.11	2.7 ± 0.77	3.6 ± 0.98
	F	74	2.3 ± 1.20	2.3 ± 1.22	2.6 ± 1.3	2.9 ± 1.10	2.5 ± 0.96	2.7 ± 0.95
	$\boldsymbol{G}$	**98**	$\mathbf{2.3 \pm 1.32}$	$\mathbf{2.1 \pm 1.32}$	$\mathbf{2.0 \pm 1.3}$	$\mathbf{1.7 \pm 0.90}$	$\mathbf{2.0 \pm 1.00}$	$\mathbf{1.9 \pm 1.05}$
	H	46	2.0 ± 1.26	2.2 ± 1.27	–	2.1 ± 1.01	2.1 ± 1.12	–
	I	73	3.0 ± 1.24	3.2 ± 1.36	–	2.7 ± 0.90	2.9 ± 0.87	–
	J	52	2.0 ± 1.08	2.9 ± 1.21	–	2.9 ± 0.92	2.3 ± 0.98	–
Marathi	A	35	4.4 ± 1.04	4.3 ± 1.00	4.3 ± 1.08	4.6 ± 0.80	4.5 ± 0.81	4.8 ± 0.64
	B	35	2.3 ± 1.22	2.7 ± 1.13	2.2 ± 1.15	2.7 ± 1.12	2.5 ± 1.22	2.2 ± 1.00
	C	26	2.3 ± 1.18	1.9 ± 0.99	1.6 ± 0.88	2.5 ± 1.05	2.7 ± 0.94	2.6 ± 0.97
	D	30	3.0 ± 1.22	3.4 ± 1.04	3.1 ± 1.21	3.0 ± 1.04	2.9 ± 0.94	2.6 ± 0.97
	E	52	2.9 ± 0.85	3.4 ± 1.21	2.7 ± 1.03	3.0 ± 0.77	3.3 ± 1.00	3.4 ± 1.03
	F	41	3.0 ± 1.11	2.8 ± 1.09	2.7 ± 1.00	3.2 ± 1.07	3.2 ± 0.97	2.9 ± 0.95
	$\boldsymbol{G}$	**69**	$\mathbf{2.5 \pm 1.16}$	$\mathbf{2.2 \pm 1.13}$	$\mathbf{2.1 \pm 1.08}$	$\mathbf{2.2 \pm 0.95}$	$\mathbf{2.2 \pm 0.99}$	$\mathbf{2.1 \pm 0.95}$
	H	35	2.1 ± 1.14	2.4 ± 1.19	–	2.9 ± 1.04	2.7 ± 1.03	–
	J	21	2.8 ± 1.14	2.4 ± 0.92	–	3.3 ± 0.97	2.9 ± 1.00	–
Tamil	A	29	4.6 ± 0.89	4.6 ± 0.71	4.2 ± 1.24	4.7 ± 0.73	4.6 ± 0.80	4.7 ± 0.75
	B	57	1.8 ± 1.05	1.9 ± 1.02	2.2 ± 1.13	2.2 ± 1.11	2.2 ± 1.10	2.2 ± 1.03
	C	49	2.2 ± 1.15	2.8 ± 1.23	2.8 ± 1.07	2.8 ± 0.99	3.3 ± 1.05	2.9 ± 1.08
	D	49	1.9 ± 1.08	2.0 ± 1.06	1.7 ± 0.93	2.6 ± 1.11	2.6 ± 1.11	2.3 ± 1.08
	E	69	2.7 ± 0.48	2.6 ± 1.28	3.3 ± 0.86	2.5 ± 0.60	3.0 ± 1.07	4.0 ± 0.91
	F	67	2.7 ± 1.21	2.5 ± 1.10	2.6 ± 1.09	3.6 ± 1.06	3.2 ± 0.97	3.3 ± 0.94
	$\boldsymbol{G}$	**47**	$\mathbf{2.3 \pm 1.14}$	$\mathbf{3.1 \pm 1.25}$	$\mathbf{2.4 \pm 1.33}$	$\mathbf{2.4 \pm 1.00}$	$\mathbf{2.3 \pm 1.06}$	$\mathbf{2.6 \pm 1.02}$
	H	57	2.6 ± 1.24	2.7 ± 1.18	–	3.0 ± 1.14	3.7 ± 0.87	–
	I	50	3.6 ± 1.11	3.4 ± 1.19	–	3.2 ± 0.97	3.0 ± 1.22	–
	J	60	2.6 ± 1.10	2.3 ± 1.21	–	2.7 ± 1.03	3.0 ± 0.89	–
Telugu	A	82	4.5 ± 0.86	3.3 ± 1.41	3.8 ± 1.31	4.8 ± 0.59	4.5 ± 0.82	4.8 ± 0.64
	B	70	2.1 ± 1.10	2.1 ± 1.02	2.1 ± 0.95	1.9 ± 0.98	1.8 ± 0.84	2.0 ± 0.96
	C	46	1.3 ± 0.82	1.4 ± 0.60	1.5 ± 0.86	2.6 ± 1.22	2.5 ± 1.12	2.5 ± 1.06
	D	60	2.0 ± 1.22	2.6 ± 1.06	2.3 ± 1.07	2.1 ± 0.96	2.5 ± 0.98	2.6 ± 0.96
	E	54	2.9 ± 0.82	2.5 ± 0.81	2.5 ± 0.90	2.8 ± 0.85	2.7 ± 0.81	2.9 ± 0.85
	F	59	2.5 ± 1.19	2.0 ± 1.01	2.2 ± 0.95	3.5 ± 0.95	2.5 ± 1.01	2.6 ± 0.98
	$\boldsymbol{G}$	**75**	$\mathbf{3.1 \pm 1.25}$	$\mathbf{2.2 \pm 1.08}$	$\mathbf{1.4 \pm 0.64}$	$\mathbf{2.1 \pm 0.97}$	$\mathbf{2.1 \pm 0.86}$	$\mathbf{2.4 \pm 0.96}$
	H	46	2.4 ± 0.93	3.4 ± 1.11	–	3.0 ± 1.06	3.0 ± 1.02	–
	I	62	4.2 ± 0.97	1.9 ± 1.04	–	2.9 ± 1.10	2.1 ± 0.86	–
	J	57	2.7 ± 1.18	2.0 ± 1.03	–	3.5 ± 0.94	2.7 ± 0.98	–

Tamil, our system $\boldsymbol{G}$ shows the lowest WER (47%). By contrast, the WER of the other four languages (Bengali, Malayalam, Marathi, and Telugu) are not good. These results suggest that, it is important to properly adjusts the front-end component for each language. The WER (98%) in Malayalam is poor results. Since there are many graphemes in the word in Malayalam, it seems that errors in the front-end component easily occur.

The speaker similarity score in our system $\boldsymbol{G}$ is neither good nor bad compared with the other systems. Bengali obtained the high speaker similarity score even though it is the high WER (76%). Therefore, a high WER or pronunciation errors scarcely affect the speaker similarity score.

In our system $\boldsymbol{G}$, the naturalness score could not get good results. Tamil obtained the low naturalness score even though it is the lowest WER. Thus, even a little word pronunciation error often affects the naturalness score.

5. Conclusions

We described a hidden Markov model (HMM)-based text-to-speech (TTS) system developed at the Nagoya Institute of Technology (NITECH) for the Blizzard Challenge 2015. In this paper, we focused on constructing a TTS system without a special knowledge of a target language. Our approach enabled a target language and multilingual TTS system construction. In large-scale subjective experiments, our system was able to achieve a high score if it is properly constructed the front-end component for each language.

Investigation of a construction criteria, construction of the multilingual speaker-independent speech recognizer (SISR) using the international phonetic alphabet (IPA), and investigation of phoneset determination approaches based on speech data [26, 27] will be future works.

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] Ethnologue. [Online]. Available:, https://www.ethnologue.com.

[2] 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.

[3] 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.

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

[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.

[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 2015. [Online]. Available:, http://www.synsig.org/index.php/Blizzard_Challenge_2015.

[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] 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.

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

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

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

[19] 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.

[20] 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.

[21] 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.

[22] 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.

[23] 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.

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

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

[26] P. K. Muthukumar and A. W. Black, "Automatic discovery of a phonetic inventory for unwritten languages for statistical speech synthesis," *Proceedings of ICASSP 2014*, pp. 2594–2598, 2014.

[27] T. Fukada, M. Bacchiani, K. Paliwal, and Y. Sagisaka, "Speech recognition based on acoustically derived segment units," *Proceedings of ICSLP 96*, vol. 2, pp. 1077–1080, 1996.