The UTokyo speech synthesis system for Blizzard Challenge 2017

Shinnosuke Takamichi, Daisuke Saito, Hiroshi Saruwatari, Nobuaki Minematsu

The University of Tokyo, 7-3-1, Hongo, Bunkyo-ku, Tokyo, 113-8656 Japan

{shinnosuke_takamichi,hiroshi_saruwatari}@ipc.i.u-tokyo.ac.jp,
{dsk_saito,mine}@gavo.t.u-tokyo.ac.jp

Abstract

This paper presents a speech synthesis system developed at the University of Tokyo (UTokyo) for the Blizzard Challenge 2017. The task of this year's challenge is the British English children's audiobook. We have developed the Deep Neural Network (DNN)-based speech synthesis system including two functions: automated bell-sound removal and an audio code. The developed system has been submitted, and the results of the large-scale subjective evaluation demonstrated the performance of our system.

Index Terms: deep neural network, DNN-based speech synthesis, audiobook, audio code

1. Introduction

To compare different speech synthesis techniques to develop a corpus-based speech synthesis system using shared datasets, Blizzard Challenge was devised in January 2005 [1] and has been held every year. This year's Blizzard Challenge has two kinds of tasks: 1) the single hub task (2017-EH1) which requires teams to build an end-to-end text-to-speech system and 2) two spoke tasks (2017-ES1 and 2017-ES2) which are designed to be accessible to the wider machine learning community. We joined only the first task (2017-EH1).

We are the joint team of System #1 Lab. and Minematsu & Saito Lab. of the University of Tokyo. Our research aimed various target towards augmented speech-based communication [2, 3, 4, 5]. To submit a speech synthesis system from our team to the Blizzard Challenge 2017, we have developed our system called the *U*Tokyo speech synthesis system. The acoustic models are Deep Neural Networks (DNNs), and we newly implemented some modules, such as an audio context and automatic bell-sound removal. The developed system has been submitted, and the results of the large-scale subjective evaluation demonstrated the performance of our system.

2. Data and task

The task of this year's Blizzard Challenge is to produce a set of voices given British English audiobook corpora. The database has approximately 6.5 hours' speech data. However, to shorten the voice building time, we used the training data used in the last year's challenge [6]. These speech data are recorded by one female speaker. The sampling rate is 44.1 kHz. In this database, there are three types of audio formats, that are MP3, WMA, and M4A. A sentence-level alignment label between text and speech is provided. Many utterances in those audiobooks are very rich in emotion with a large number of onomatopoetic words. They also include non-speech sounds such as a bell sound. The testing transcriptions include texts collected from audiobooks, news and Semantically Unpredictable Sentences (SUS). Therefore, different sets of audio are required to be synthesized.

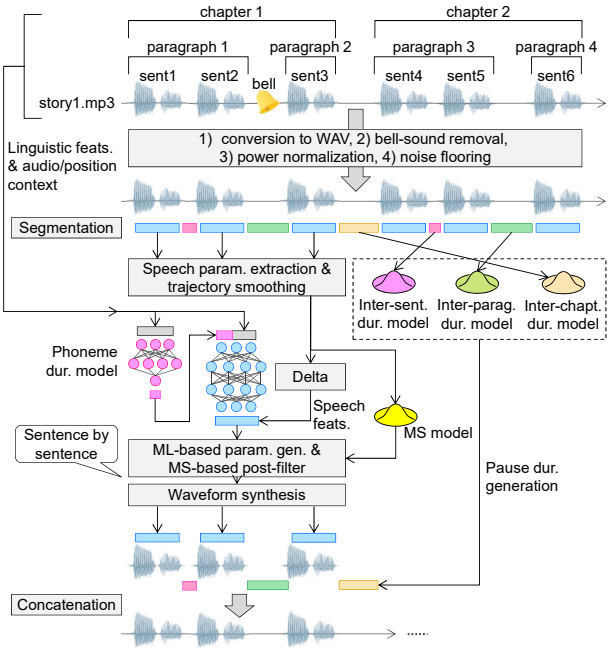

Figure 1: *Overview of the UTokyo speech synthesis system. We train three kinds of models: 1) acoustic models and phoneme duration models, 2) Modulation Spectrum (MS) models, and 3) pause duration models consisting of inter-sentence, inter-paragraph, and inter-chapter duration models. "sent*" is the sentence ID. ML indicates Maximum Likelihood.*

3. UTokyo speech synthesis system

Our system mainly consists of 4 modules: speech processing, text processing, training, and speech synthesis modules. The overview is shown in Fig. 1.

3.1. Speech processing module

This module performs audio data preprocessing and speech parameter extraction. First, audio formats (MP3, WMA and M4A) of the training data was converted to the RIFF WAV format, using the sox command that is a command-line audio processing tool for Linux. In the audio data, bell sounds are included at the timing of the page turning. Since the sound is expected to be completely the same as all the time, we expect that it can be removed by a simple approach such as spectral matching. One of bell sounds was initially extracted as the example, and then, using the spectral matching, the bell sounds of the audio data were detected and removed automatically. Finally, we

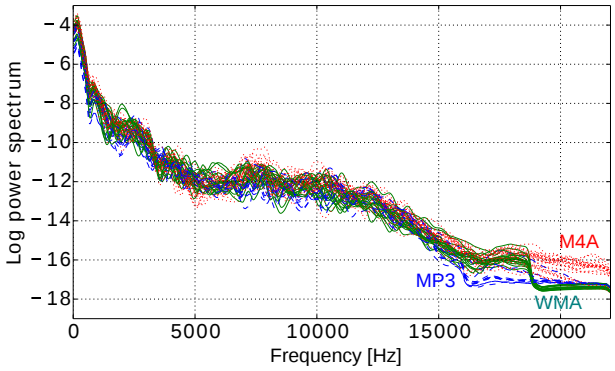

Figure 2: *An example of log-scaled power spectra. Each line is the spectrum averaged in each audiobook story. The audio format used in this year's challenge changes the spectra at the higher frequency bands.*

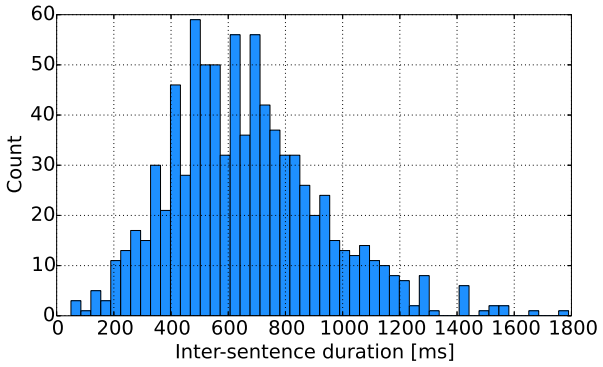

Figure 3: *Histogram of inter-sentence duration. The Gaussian distribution is used to model this histogram.*

performed power normalization and noise flooring to the audio data.

After preprocessing, speech parameters are extracted from the audio data, using the WORLD analysis-synthesis system [7, 8]. The speech parameters were 0th-through-59th mel-cepstral coefficients, 4-band band aperiodicity, continuous log-scaled F_0, and unvoiced/voiced labels. The shift length was 5 ms. Speech parameter trajectory smoothing [9] was performed to improve the training accuracy, using low-pass filters with 50 Hz (mel-cepstral coefficients) and 10 Hz cutoff modulation frequency. The trajectory smoothing removes the detailed structures of speech parameters that are negligible for speech perception.

3.2. Text processing module

This module extracts linguistic features and makes a contextual vector for conditioning acoustic models. 387-dimensional sentence-level linguistic features, which are often used for reading-style speech synthesis, were extracted using Flite [10]. We added 3-dimensional binary style contexts (0: reading, 1: speaking, 2:mixed). The style context is determined by whether or not the sentence is enclosed by apostrophes. Sentiment analysis was not used. The position contexts, such as the position of the current sentence in the current paragraph, were also used. As an audio context, we used the 3-dimensional binary audio code. As shown in Fig. 2, the format of the original audio data changes spectral parameters, especially at the higher frequency components. The audio code (0: M4A, 1: MP3, 2: WMA) supports such a global change of the speech parameters.

3.3. Training module

This module trains statistical models to predict speech parameters from the input transcription. We built three kinds of the statistical models: acoustic models, Modulation Spectrum (MS) models, and pause duration models.

3.3.1. Acoustic models and phoneme duration modules

We built the standard DNN-based speech synthesis [11] using speech segments. The DNN architectures of the acoustic models were Feed-Forward neural networks that include 3 512-unit Rectified Linear Unit (ReLU) hidden layers [12] and a 196-unit linear output layer. Architectures of the phoneme duration model were the same, but the number of units of the output layer was 1. The linguistic features included features described in Sec. 3.2 and 3-dimensional phoneme duration contexts. The speech features included static, delta, and delta-delta of speech parameters. The contextual features and speech features were normalized to have zero-mean unit-variance, and 95% of the silence frames were removed from the training data. 2400 utterances were used for initial training, and 1900 utterances having higher training accuracy were selected for retraining the acoustic models.

3.3.2. Modulation Spectrum (MS) models

To enhance generated speech parameters, we trained MS models using non-silence segments. The zero-mean segment-level MSs were calculated, and the distribution was modeled with the context-independent Gaussian distribution. Similarly, a distribution of the MSs of synthetic speech is also modeled in the same manner.

3.3.3. Pause duration models

To predict pause duration between sentences, paragraphs, and chapters, we trained three kinds of pause duration models: inter-sentence, inter-paragraph, inter-chapter duration models. Using text-audio alignments, the pause duration was extracted. The distribution was modeled with the context-independent model. Since the distribution seems to be unimodal and symmetric as shown in Fig. 3, we utilized the Gaussian distribution to fit it.

3.4. Synthesis module

Speech parameters were generated sentence by sentence, and the waveform was synthesized. Then, the inter-sentence, inter-paragraph, and inter-chapter duration were generated. Finally, they were simply concatenated to synthesize a waveform of an audiobook.

3.4.1. Speech parameter generation and post-processing

1-dimensional phone duration was predicted at each phoneme and was encoded to the 3-dimensional duration contexts. The speech features were predicted using the duration contexts. Maximum Likelihood (ML)-based generation algorithm [13]

was performed to generate mel-cepstrum, continuous log-scaled F_0, unvoiced/voiced labels, and band-aperiodicity from the predicted features. MS-based post-filtering [14] setting the emphasis coefficient to 0.85 was performed to the generated mel-cepstral coefficients and continuous log-scaled F_0. Since the post-filtering causes some over-emphasis, we finally applied the trajectory smoothing to the post-filtered speech parameters. The WORLD synthesis system was used to synthesize the sentence-level speech waveforms.

3.4.2. Pause duration generation

The pause duration is determined using the pause duration models. The inter-sentence and inter-paragraph duration were randomly sampled from their Gaussian distribution trained in advance. The ML estimate was used for determining inter-chapter duration from its Gaussian distribution. The final speech was simply synthesized by concatenating sentence-level speech described in Section 3.4.1 and pause duration described here.

4. Experimental evaluation

4.1. Experimental settings

Our designated system identification letter is 'N.' System A is natural speech. System B is the Festival benchmark system based on unit-selection. System C is the HTS benchmark. System D is a DNN benchmark using Merlin toolkit. Others are participants' systems. The subjects who are involved in the listening test are paid listeners, speech experts, and online volunteers. This year, there are mainly four sections to evaluate, which are a paragraph test, a naturalness test, a similarity test, and a SUS (Semantically Unpredictable Sentences) test. For the paragraph test, there are seven kinds of tests to evaluate different aspects of synthesized paragraphs, namely overall impression, pleasantness, speech pauses, stress, intonation, emotion, and listening effort.

4.2. Results and analysis

Because of the limited space, we show only 1) mean opinion scores on overall impression of audiobook paragraphs in Fig. 4, 2) naturalness in Fig. 5, 3) speaker similarity in Fig. 6, and 4) word error rates using SUS sentences in Fig. 7.

Totally, our results were not good. Here, we discuss the defects. The two main defects are poor performance of text-audio alignment and poor contexts. In this challenge, we used our tool for the alignment, but it is not totally accurate. Therefore, the phonetic property was significantly lost even in the closed data. Also, because we did not use sentiment analysis and other related techniques to distinguish reading style and emotional style, the prosody of the synthetic speech became unnatural.

5. Conclusion

We introduced the UTokyo speech synthesis system for Blizzard Challenge 2017. The results of the listening test for our system were not good, but we have found many interesting problems that we should have attacked.

6. Acknowledgements

Part of this work was supported by SECOM Science and Technology Foundation, and JSPS KAKENHI Grant Number 16H06681.

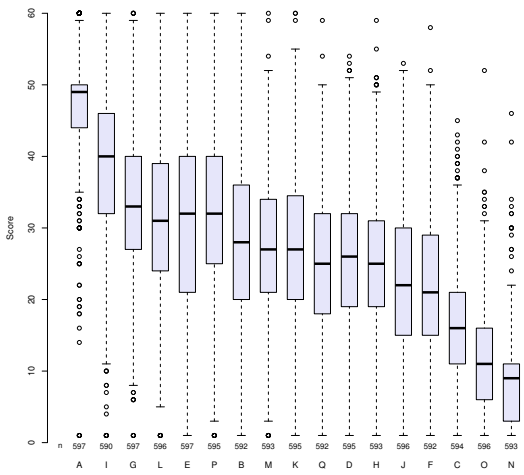

Figure 4: *Mean opinion scores (overall impression of audiobook paragraphs). Results by all listeners were included.*

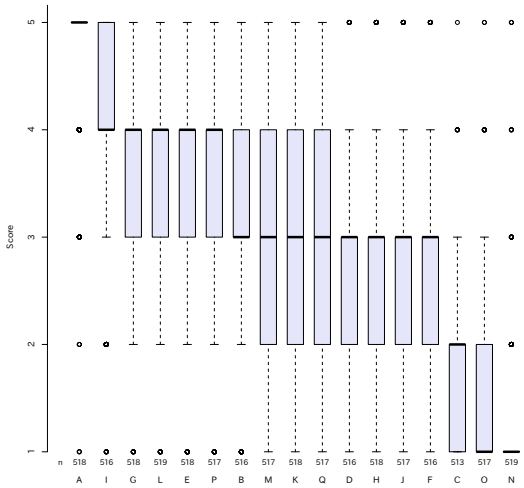

Figure 5: *Mean opinion scores (naturalness of synthetic speech). Results by all listeners were included.*

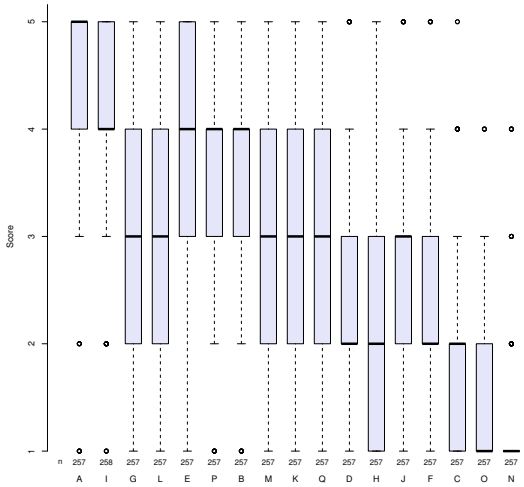

Figure 6: *Mean opinion scores (similarity of synthetic speech to original speaker). Results by all listeners were included.*

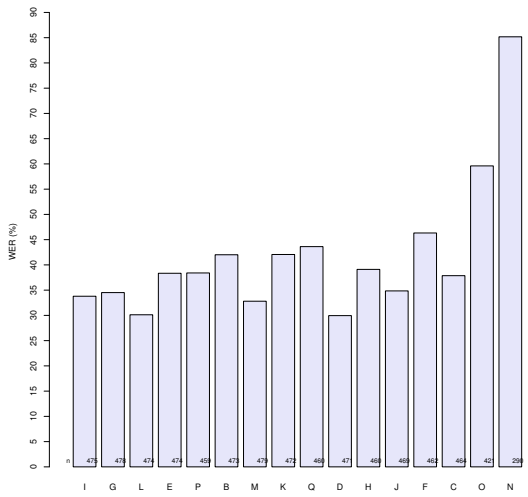

Figure 7: *Word error rates using semantically unpredictable sentences. Results by all listeners were included.*

7. References

[1] A. W. Black and K. Tokuda, "The blizzard challenge-2005: Evaluating corpus-based speech synthesis on common datasets," in *Proc. INTERSPEECH*, Lisbon, Portugal, Sep. 2005.

[2] S. Takamichi, K. Tomoki, and H. Saruwatari, "Sampling-based speech parameter generation using moment-matching network," in *Proc. INTERSPEECH*, Stockholm, Sweden, Aug. 2017.

[3] Y. Saito, S. Takamichi, and H. Saruwatari, "Training algorithm to deceive anti-spoofing verification for DNN-based speech synthesis," in *Proc. ICASSP*, Orleans, U.S.A., Mar. 2017.

[4] S. Shi, Y. Kashiwagi, S. Toyama, J. Yue, Y. Yamauchi, D. Saito, and N. Minematsu, "Automatic assessment and error detection of shadowing speech: Case of English spoken by Japanese learners," in *Proc. INTERSPEECH*, Sep. 2016, pp. 3142–3146. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2016-915

[5] H. Uchida, D. Saito, and N. Minematsu, "Prediction of the articulatory movements of unseen phonemes of a speaker using the speech structure of another speaker," in *Proc. INTERSPEECH*, San Francisco, U.S.A., Sep. 2016, pp. 450–454. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2016-1138

[6] Y. Zhao, X. You, D. Saito, and N. Minematsu, "The UTokyo system for Blizzard Challenge 2016," in *Proc. Blizzard Challenge Workshop*, Sep. 2016.

[7] M. Morise, F. Yokomori, and K. Ozawa, "WORLD: a vocoder-based high-quality speech synthesis system for real-time applications," *IEICE transactions on information and systems*, vol. E99-D, no. 7, pp. 1877–1884, 2016.

[8] M. Morise, "D4C, a band-aperiodicity estimator for high-quality speech synthesis," *Speech Communication*, vol. 84, pp. 57–65, 2016.

[9] S. Takamichi, K. Kobayashi, K. Tanaka, T. Toda, and S. Nakamura, "The NAIST text-to-speech system for the Blizzard Challenge 2015," in *Proc. Blizzard Challenge workshop*, Berlin, Germany, Sep. 2015.

[10] "CMU Flite: a small, fast run time synthesis engine http://www.festvox.org/flite/."

[11] H. Zen, A. Senior, and M. Schuster, "Statistical parametric speech synthesis using deep neural networks," in *Proc. ICASSP*, Vancouver, Canada, May 2013.

[12] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. AISTATS*, Lauderdale, U.S.A., Apr. 2011, pp. 315–323.

[13] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," in *Proc. ICASSP*, Istanbul, Turkey, June 2000, pp. 1315–1318.

[14] S. Takamichi, T. Toda, A. W. Black, G. Neubig, S. Sakti, and S. Nakamura, "Postfilters to modify the modulation spectrum for statistical parametric speech synthesis," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 4, pp. 755–767, 2016.