The NTUT+III's Chinese Text-to-Speech System for Blizzard Challenge 2019

Yuan-Fu Liao[1] *and Cheng-Hung Tsai*[2]

[1]National Taipei University of Technology, Taipei, Taiwan
[2]Institute for Information Industry, Taipei, Taiwan

[1]yfliao@ntut.edu.tw

Abstract

To build a Chinese text-to-speech (TTS) system, this work focused on developing a Chinese natural language processing (NLP) frontend from scratch for linguistic feature extraction. For the backend, an HMM/DNN-based Speech Synthesis System (HTS)-based speech synthesis system developed for last year's challenge was simply adopted due to resource limitation. Although, the performance of our system is not good enough on naturalness and similarity measurement, however its pinyin error rates (with and without tone, PTER and PER) may be not too bad, i.e. its voice is still intelligible. In any way, we now have a complete Chinese TTS with both frontend and backend for further performance improvement.

Index Terms: natural language processing, speech synthesis, HTS.

1. Introduction

This paper describes our Chinese Text-to-Speech system submitted to Blizzard Challenge 2019 [1]. In this year's challenge, a single speaker corpus with in total 8 hours of Mandarin speech data was provided. Since, the data was from a very popular internet talk show in China, the task is therefore to build a voice that is suitable for expressive TTS.

Unlike previous Blizzard challenge [2], where reference linguistic feature, especially, the "label", files were given along with the speech data, in this year, only 480 MP3 and their corresponding transcription files were provided. In other words, all participants have to build not only the speech synthesis backend but also the Chinese text analysis frontend.

To fulfill the requirements of challenge, this work focused on developing a Chinese NLP frontend from scratch for linguistic feature extraction. On the other hand, an HTS-based speech synthesis [3], [4] backend developed in last year's challenge [5] was simply adopted due to time and resource limitation.

This system is our first complete Chinese TTS with both NLP frontend and speech synthesis backend. In the following sections, the Chinese NLP frontend, automatic labelling of data, speech synthesis backend and finally the official evaluation results will be described in details.

2. System Configuration

2.1. Chinese Natural Language Processing Frontend

One feature of this talk show corpus is that the given text was more like in the written form of speech, especially all punctuation marks were well annotated. A typical text example of the corpus is as follows:

- 前一段时间我看到一篇文章，说上世纪六零年代的“大逃港”时，香港人真是好，全民动员救助逃过去的吃不饱饭的大陆人，可见香港人的精神境界有多高。

Therefore, not only sentences-level but also supra-sentential context label sets could be utilized to take advantage of the fact that the data were not simply isolated sentences.

On the other hand, there are also many English words in the corpus that may be an issue, for example:

- 把「得到」**App** 里的一些专栏内容，拿出来变成了**20**个精品课，一次推给大家。
- 当年谷歌和 **Facebook** 推出精准的互联网广告，企业界一片欢呼。

So, the NLP frontend have to deal with mixed Chinese and English sentences.

In the following subsections, text normalization, word segmentation, part-of-speech (POS) tagging and grapheme-to-phoneme (G2P) were first done. Then, prosodic breaks were added according to the word segmentation results and punctuation marks. Intonation cues were also applied to distinguish between different types of sentences. A typical example (in fact, it is the test case of our text analysis program) of the text analysis outputs in all essential steps are shown in Table 1.

Table 1. Typical outputs of essential text analysis procedures.

Procedure	Output
Original	中華電信、大 4G，提供：130MHz！頻寬最大？唯一4CA 四頻聚合。
Text Normalization	中華電信、大四 G，提供：一百三十 MHz！頻寬最大？唯一四 CA 四頻聚合。
English	中華電信、大四克，提供：一百三十艾马艾尺贼德！頻寬最大？唯一四西诶四頻聚合。
Word Segmentation	中華電信 、 大四 克 ， 提供 ： 一百三十 艾马艾 尺 贼 德 ！ 頻寬 最大 ？ 唯一 四西 诶 四頻 聚合。
POS	中華電信/n 、大四/m 克/m ，/w 提供/v ：/w 一百三十/m 艾马艾/n 尺/q 贼/n 德/j ！/w 頻/a 寬/a 最大/a ？/w 唯一/b 四西/n 诶/e 四頻/d 聚合/v 。/w
Break and Intonation	中華電信/n/q #1 大四/m/q 克/m/q #3 提供/v/q #3 一百三十/m/q 艾马艾/n/q 尺/q/q 贼/n/q 德/j/q #5 頻/a/e 寬/a/e 最大/a/e #6 唯一/b/d 四西/n/d 诶/e/d 四頻/d/d 聚合/v/d #4

Finally, a hierarchical linguistic tree structure of the test input text as show in Table 2 was constructed to extract linguistic features in six different levels including (1) paragraph, (2) sentence, (3) clause, (4) phrase, (5) word and (6) syllable (in pinyin).

Table 2. The hierarchical linguistic tree structure of a test input text. There are six break levels including (1) paragraph, (2) sentence, (3) clause, (4) phrase, (5) word and (6) syllable (in pinyin).

<table>
<tr><td>0</td><td colspan="32">中華電信、大 4G，提供：130MHz！頻寬最大？唯一 4CA 四頻聚合。</td></tr>
<tr><td>1</td><td colspan="32">中華電信 #1 大四克 #3 提供 #3 一百三十艾马艾尺贼德 #5 頻寬最大 #6 唯一四西诶四頻聚合 #4</td></tr>
<tr><td>2</td><td colspan="19">中華電信 #1 大四克 #3 提供 #3 一百三十艾马艾尺贼德 #5</td><td colspan="4">頻寬最大 #6</td><td colspan="9">唯一四西诶四頻聚合 #4</td></tr>
<tr><td>3</td><td colspan="7">中華電信大四克</td><td colspan="2">提供</td><td colspan="10">一百三十艾马艾尺贼德</td><td colspan="4">頻寬最大</td><td colspan="9">唯一四西诶四頻聚合</td></tr>
<tr><td>4</td><td colspan="4">中華電信</td><td colspan="3">大四克</td><td colspan="2">提供</td><td colspan="10">一百三十艾马艾尺贼德</td><td colspan="4">頻寬最大</td><td colspan="9">唯一四西诶四頻聚合</td></tr>
<tr><td>5</td><td colspan="4">中華電信</td><td colspan="2">大四</td><td>克</td><td colspan="2">提供</td><td colspan="4">一百三十</td><td colspan="3">艾马艾</td><td>尺</td><td>贼</td><td>德</td><td>頻</td><td>寬</td><td colspan="2">最大</td><td colspan="2">唯一</td><td colspan="2">四西</td><td>诶</td><td colspan="2">四頻</td><td colspan="2">聚合</td></tr>
<tr><td>6</td><td>zhongl</td><td>hua2</td><td>dian4</td><td>xin4</td><td>da4</td><td>sic4</td><td>ke4</td><td>ti2</td><td>gong1</td><td>yi4</td><td>bai3</td><td>san1</td><td>shih2</td><td>ai4</td><td>ma3</td><td>ai4</td><td>chih3</td><td>zei2</td><td>de2</td><td>pin2</td><td>kuan1</td><td>zui4</td><td>da4</td><td>wuei2</td><td>yi1</td><td>sic4</td><td>xi1</td><td>ei2</td><td>sic4</td><td>pin2</td><td>jv4</td><td>he2</td></tr>
</table>

2.1.1. *Text Normalization*

Since the given text of the talk show corpus is in the written form of speech, they have to be first converted into spoken language for speech synthesis [6]. To this end, the "cn-text-normalizer" [7] toolkit was modified to convert the numbers, units, dates, formulas, English words and some symbols in the input text. The second and third rows of Table 1 show the text-normalization results of a test case.

It is worth noting that all English words in the original text were first spell out and then each alphabet were further replaced by a set of Chinese characters according to the table of Chinese spelling of the English alphabet [8]. For example, "MHz" will be converted into "艾马 艾尺 贼德". Because, in China, letters of the English alphabet are pronounced somewhat differently from the original, as they are adapted to the phonetics of the Chinese language. This is of course just a temporary approach. Since, there was too few English speech data available in the talk show corpus.

2.1.2. *Word Segmentation and POS tagging*

For Chinese word segmentation and POS tagging [9], the "jieba" [10] toolkit was simply adopted. 26 different types of POS (including punctuation marks) were annotated. Typical segmentation and POS tagging results were shown in the fourth and fifth row of Table 1.

2.1.3. *Break and Intonation*

The punctuation marks given in the input text were directly utilized to insert breaks [11]. Six levels of breaks as shown in Table 2 were used, including (1) paragraph, (2) sentence, (3) clause, (4) phrase, (5) word and (6) syllable. A typical processing result was shown in the last row of Table 1.

Moreover, four types of intonations were considered including declarative, interrogative, imperative, exclamatory sentences. They were tagged as "*/d", "*/q", "*/i" and "*/e", respectively, as also shown in the last row of Table 1.

2.1.4. *Phonemes and Lexicon*

Mandarin is a mono-syllable and tonal language [12] where each Chinese character is pronounced as a tonal syllable. Table 3 shows the phonetic structure of Chinese syllables. Although, there are more than 20,000 characters, there are in total only about 410 toneless and 1300 tonal syllables. Therefore, sub-syllable units, such as *initials* and *finals*, are popular for both Mandarin speech recognition and synthesis. In this study, 23 initials and 39 finals as shown in Table 3 were used.

Table 3. The phonetic structure of Mandarin syllables that consists of an optional initial (null or consonants), followed by a final (glide+vowel+nasal, glide and nasal are optional).

Syllable		
Initial	*Final*	*Tone*
b, p, m, f, d, t, n, l, g, k, h, j, q, x, zh, ch, sh, r, z, c, s, y, w	a, o, e, ea, i, u, v, ic, ih, er, ai, ei, ao, ou, ia, ie, ua, uo, ve, iao , iou, uai, uei, an, ian, uan, van, en, in, uen, vn, ang, iang, uang, eng, ing, ueng, ong, iong	1, 2, 3, 4, 5

Moreover, after word segmentation, the "pypinyin" [10] toolkit was adopted to find the pronunciation of each word according to the Hanyu Pinyin standard [13] (as shown in the bottom of Table 2). The "pypinyin" toolkit has a built-in lexicon with 42,963 words collected from internet.

2.1.5. *Linguistic Features*

According to the constructed linguistic trees, the linguistic features were extracted and the corresponding question set used for context-dependent phones clustering was composed as listed in Table 4.

It is worth noting that supra-segmental linguistic features across utterance were considered here. In brief, most features were number, position and break-related cues, such as the number and forward and backward positions of phone, syllable, word and sentence in a syllable, word, phrase and paragraph, respectively. However, breaks and intonation information were also utilized to build a voice that is suitable for expressive TTS.

Table 4: Hierarchical structure of linguistic features for acoustic modeling.

Layer	Feature/Question
Phone	names and types of current and surrounding phones (5-gram); number and forward and backward position of a phone in a syllable
Syllable and Tone	number and forward/backward position of a syllable in a word; names of current and surrounding tones (3-gram)
Word	part-of-speech (POS) of current and surrounding words; the number and forward and backward position of a word in a phrase
Phrase	number and forward and backward position of a phrase in a sentence
Sentence	number and forward and backward position of sentence a in paragraph; intonations of the current and surrounding sentence (3-gram)

2.2. Forced-Alignment

Automatic labelling of the data was implemented using Montreal forced aligner [14], [15] which is a command line utility for forced alignment using Kaldi [16], [17] toolkit.

In order to achieve more precise segmentation, a speaker-dependent triphone model was trained using the whole talk show corpus. Besides, 39 dimensional Mel-Frequency Cepstral Coefficients (MFCCs) [18] were utilized to train the underlying HMMs.

2.3. Speech synthesis Backend

The speech synthesis part mainly followed our last year's approach. In other words, it was the same HTS-based system as in [5].

2.3.1. Acoustic Features

The MP3 files in the talk show corpus was first decoded using the "sox" toolkit [19] and then down-sampling from 24,000 to 22,050 Hz. Then speech features were extracted every 5 ms with 25 ms windows size. 49-order mel-generalized cepstrum (MGC) [20] coefficients and fundamental frequency (F0) were extracted as the spectral and excitation parameters. Moreover, the first and second order derivative features were also computed to form a 150-dimentional feature vector for each speech frame.

2.3.2. Acoustic Modeling

DNN-based acoustic models were trained to establish a complex linguistic to output acoustic features mapping function. A trajectory training criterion considering global variance (GV) [21] was also applied.

In the DNN models, three hidden layers that had 2048 neurons with rectified linear unit (RELU) activation functions. The post-filter options were applied.

3. Training Procedure

The voice building steps of our system are briefly showed in Fig. 1. Especially, the deterministic annealing expectation and maximization (DAEM) [22] training algorithm was utilized to well initialize the HMM-based acoustic modeling. The number of iterations for DAEM were experimentally set to 10. The training iteration for the DNN and Trajectory models were set to 100 and 50, respectively.

Since the training procedure is very time-consuming, some scripts and commands in HTS 2.3.2 recipe were modified in order to be parallelly executed using multiple CPUs at the same time including the (1) data preparation, (2) acoustic model training and (3) context-dependent phone clustering procedures.

The modification was based on (1) the multi-threading features of make command, (2) the integration of HTS commands (HCopy, HERest, HVite and HSMMAlign) and Sun Grid Engine (SGE) queueing system [23] and (3) the feature of HHEd's "JM" command. Basically, the operations of these steps were:

- Divide the list of data files
- Send the divided jobs to different processors
- Finish all jobs
- Combine the results

For detail information of the parallel processing, please check our last year's system [5].

By this way, it required about 3~4 days to train the HMMs and 3~4 more days to complete the DNN training using a computer cluster with 5 CPUs (Xeon E5-2650 v3 @ 2.30GHz, 10 cores, 20 threads) and 4 K80 GPUs.

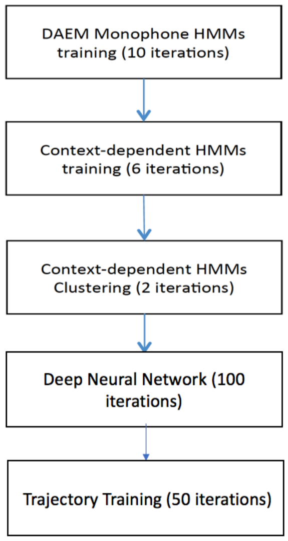
DAEM Monophone HMMs training (10 iterations)
Context-dependent HMMs training (6 iterations)
Context-dependent HMMs Clustering (2 iterations)
Deep Neural Network (100 iterations)
Trajectory Training (50 iterations)

Figure 1: The flowchart of the HMM- and DNN-based voice building procedure using HTS version 2.3.2.

4. Official Evaluation Results

In all the following official evaluation results (Fig. 2~4), our system was indicated by the letter "U".

First of all, the overall performance of our system is, of course, not good enough. Fig. 2 and 3 show the official evaluation results on naturalness and similarity reported by all listeners. However, it is interesting that our system's performance on speech intelligibility is not too bad. Fig. 4 and shows the official pinyin error rate (with and without PTER and PER in %), on speech intelligibility reported by paid listeners.

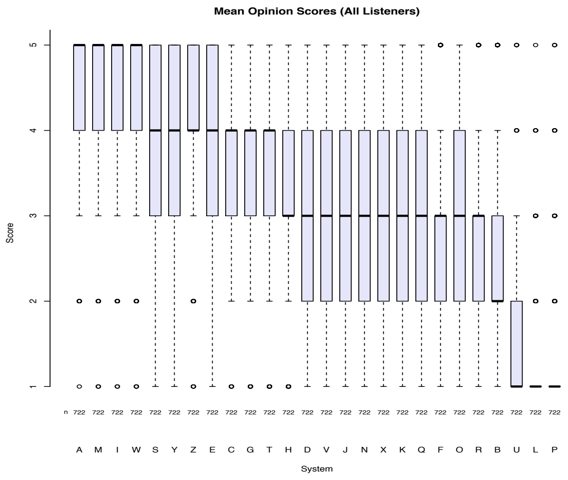

Figure 2: *Naturalness evaluation results (in MOS) reported by all listeners.*

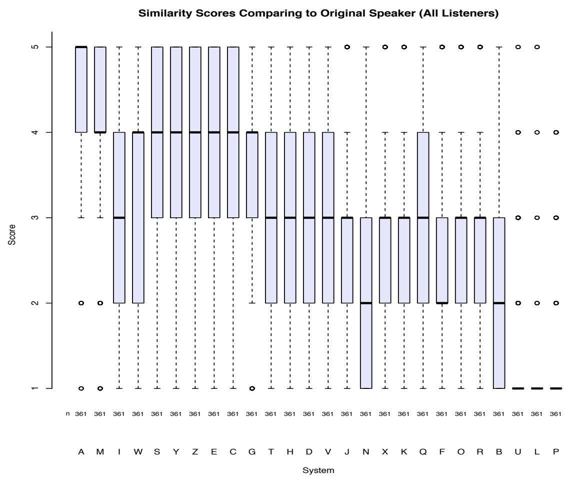

Figure 3: *Similarity evaluation results (in MOS) reported by all listeners.*

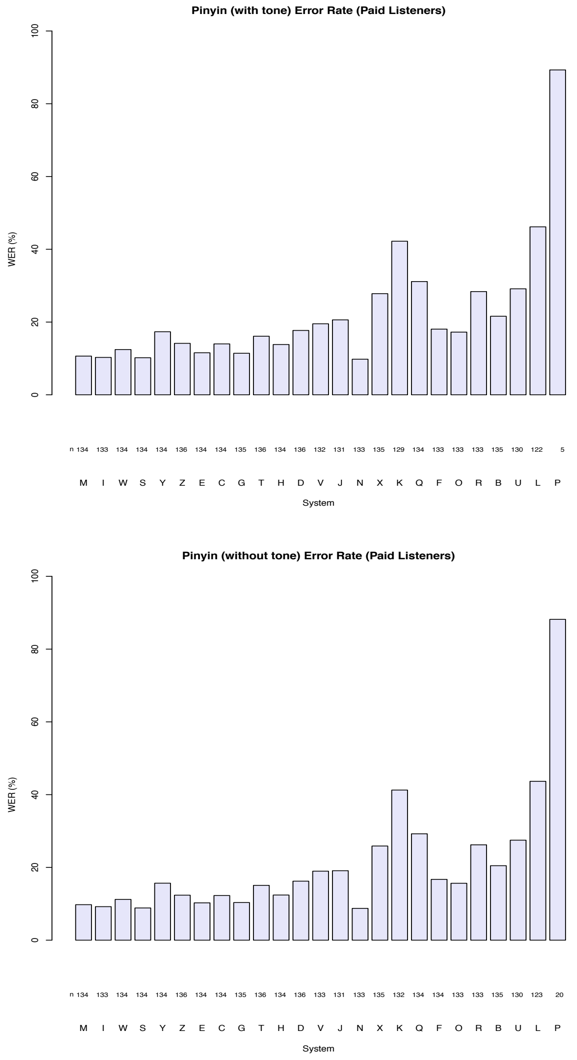

Figure 4: *Speech intelligibility evaluation results on pinyin error rates, (with and without tone, i.e., PTER and PER in %) reported by paid listeners.*

5. Conclusions

This system is our first complete Mandarin speech synthesis system with both Chinese NLP and speech synthesis modules. The performance of our system is not good enough on naturalness and similarity evaluation, however its PTER and PER may be not too bad, i.e. its voice is still intelligible.

After some analysis, it is found that the problem may come from improper text-normalization and phone segmentation. Further investigations are still ongoing. In any way, we now have a complete Chinese TTS with both frontend and backend for further performance improvement.

6. Acknowledgements

This study is conducted under the “System-of-systems driven emerging service business development Project (4/4)” of the Institute for Information Industry which is subsidized by the Ministry of Economic Affairs of the Republic of China and is partially supported by the Ministry of Science and Technology of the Republic of China under the contract number 106-3011-F-027-003, 107-3011-F-027-003, 107-2221-E-027-102, 108-2221-E-027-067 and 107-2911-I-027-501 and 108-2911-I-027-501.

7. References

[1] “Blizzard Challenge 2018 - SynSIG.” [Online]. Available: https://www.synsig.org/index.php/Blizzard_Challenge_2018. [Accessed: 15-Jul-2018].

[2] M. Fraser and S. King, “The Blizzard Challenge 2008,” *Proc. Blizzard Chall. Work. 2008*, pp. 1–12, 2008.

[3] K. Hashimoto, K. Oura, Y. Nankaku, and K. Tokuda, “Trajectory training considering global variance for speech synthesis based on neural networks,” in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2016, vol. 2016-May, pp. 5600–5604.

[4] Z. Wu and S. King, “Improving Trajectory Modelling for DNN-Based Speech Synthesis by Using Stacked Bottleneck Features and Minimum Generation Error Training,” *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 24, no. 7, pp. 1255–1265, 2016.

[5] Y. Liao, Y. Chai, and C. Tsai, “The NTUT ’ s Text-to-Speech System for Blizzard Challenge 2018,” 2018.

[6] Q. Zhang, H. Chen, and X. Huang, “Chinese-English mixed text normalization,” *WSDM 2014 - Proc. 7th ACM Int. Conf. Web Search Data Min.*, pp. 433–442, 2014.

[7] M. Song, “open-speech/cn-text-normalizer: A python module that convert chinese written string to read string,” *GitHub*. [Online]. Available: https://github.com/open-speech/cn-text-normalizer/. [Accessed: 12-Aug-2019].

[8] A. Garnaut and T. Lu, *Chinese respelling of the English alphabet*. GitHub.

[9] X. Zheng, H. Chen, and T. Xu, “Deep learning for Chinese word segmentation and POS tagging,” *EMNLP 2013 - 2013 Conf. Empir. Methods Nat. Lang. Process. Proc. Conf.*, no. October, pp. 647–657, 2013.

[10] J. Sun, “Jieba: 结巴中文分词,” *GitHub*. [Online]. Available: https://github.com/fxsjy/jieba. [Accessed: 12-Aug-2019].

[11] C. Y. Chiang, Y. P. Hung, H. Y. Yeh, I. Bin Liao, and C. M. Pan, “Punctuation-generation-inspired linguistic features for Mandarin prosody generation,” *Eurasip J. Audio, Speech, Music Process.*, vol. 2019, no. 1, 2019.

[12] J. Cikoski, “A lexicon of classical chinese,” vol. II.

[13] *ISO 7098:1982 – Documentation – Romanization of Chinese.* .

[14] Z. Wu, O. Watts, and S. King, “Merlin : An Open Source Neural Network Speech Synthesis System,” *9th ISCA Speech Synth. Work.*, pp. 218–223, 2016.

[15] MontrealCorpusTools, “Montreal-Forced-Aligner,” *GitHub*. [Online]. Available: https://github.com/MontrealCorpusTools/Montreal-Forced-Aligner. [Accessed: 12-Aug-2019].

[16] D. Povey *et al.*, “The Kaldi Speech Recognition Toolkit,” in *ASRU 2011*, 2011.

[17] “kaldi-asr/kaldi: The official location of the Kaldi project,” *GitHub*. [Online]. Available: https://github.com/kaldi-asr/kaldi. [Accessed: 13-Aug-2019].

[18] P. Ghahremani, B. Babaali, D. Povey, K. Riedhammer, J. Trmal, and S. Khudanpur, “A Pitch Extraction Algorithm Tuned for ASR,” in *ICASSP 2014*, 2014.

[19] “SoX - Sound eXchange | HomePage.” [Online]. Available: http://sox.sourceforge.net/. [Accessed: 16-Jul-2018].

[20] S. Imai, T. Fukada, K. Tokuda, T. Kobayashi, S. Imai, and C. C. Llaguno, “Cepstral analysis synthesis on the mel frequency scale, and an adaptative algorithm for it.,” *Synthesis (Stuttg).*, 2008.

[21] T. Toda and S. Young, “Trajectory training considering global variance for HMM-based speech synthesis,” *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, no. April 2009, pp. 4025–4028, 2009.

[22] W. T. De Bary, W. Chan, and R. J. Lufrano, *Sources of Chinese tradition, volume II : from 1600 through the twentieth century*. Columbia University Press, 2000.

[23] B. Lee, “Parallel Processing of the HTK Commands.” [Online]. Available: http://www.ifp.illinois.edu/~bowonlee/research/cluster/HTK_parallel.htm. [Accessed: 15-Jul-2018].