SIMULTANEOUS SPEECH RECOGNITION AND ACOUSTIC EVENT DETECTION USING AN LSTM-CTC ACOUSTIC MODEL AND A WFST DECODER

Hiroshi Fujimura, Manabu Nagao*†, Takashi Masuko*

Corporate Research & Development Center, Toshiba Corporation

{hiroshi4.fujimura, manabu.nagao, takashi.masuko}@toshiba.co.jp

ABSTRACT

This paper proposes a new approach to simultaneous speech recognition and acoustic event detection of spontaneous speech based on one-pass decoding without rescoring. In this approach, an LSTM (long short-term memory) acoustic model outputs probabilities of acoustic event symbols, such as filler symbols and word fragment symbols, as well as probabilities of phonetic symbols. Then a WFST (weighted finite state transducer) decoder detects fillers based on a filler confidence score calculated by the ratio of the number of filler symbols to the number of phonetic symbols in each word. The WFST decoder also detects word fragments using a phonetic symbol loop followed by a path accepting the word fragment symbol, which is attached to a WFST of a lexicon. Experimental results show that precision and recall rates of filler detection can be controlled by the filler confidence score, and word fragments can be detected without registering all possible word fragments to the lexicon.

Index Terms— Speech recognition, WFST, detection, filler, word fragment

1. INTRODUCTION

Spontaneous speech often includes a lot of fillers and word fragments such as "um", "ah", and "thi-this". They not only lead significant performance deterioration of speech recognition [1], but also decrease readability of real-time captioning for human. To solve these problems, speech recognition systems need to detect and selectively remove fillers and word fragments. Since many applications of spontaneous speech recognition, such as speech-to-speech translation and speech dialogue systems, require real-time computation, it is necessary to to detect fillers and word fragments with very low latency.

Some techniques have been proposed [2, 3, 4] to detect word fragments using subword language models and confusion networks without modeling acoustic aspects of word fragments. However, since fillers and word fragments have distinct acoustic features, it is better to exploit such acoustic features for filler and word fragment detection. In [5], fillers are independently modeled using acoustic-prosodic features, and the model is applied to calculation of filler scores of filler hypotheses during second-pass decoding, which prevents real-time decoding. In [6], word fragments are directly modeled by adding a fragment tag to phonemes composing word fragments, and phoneme models with the fragment tag are used as garbage models. We also proposed a technique to model filler and word fragment symbols in addition to phonetic symbols as outputs of an LSTM-CTC acoustic model [7]. In this technique, fillers and word fragments were registered to a lexicon and filler and word fragment symbols were added to their pronunciation. However, it is impracticable to register all possible word fragments to the lexicon. To overcome this problem, in this paper, we propose a new decoding technique that can detect fillers and word fragments using a conventional lexicon.

End-to-end training of an acoustic model and a language model has recently been proposed [8, 9, 10]. Many methods use a bi-directional LSTM model and an attention mechanism. However, it is not suitable for real-time applications because the standard attention mechanism requires whole input speech. In addition, some methods use both end-to-end models and a conventional language models by means of a conventional decoder [8, 11]. Thus, WFST decoder [12] is still important for an efficient decoding. Our proposed decoder discriminates fillers and word fragments from normal words using a WFST that can accept filler and word fragment symbols. For fillers, the WFST is generated from a conventional lexicon and a language model, and the decoder determines fillers by checking symbols on paths of a lattice. In addition, the decoder detects word fragments using paths for word fragments added to a WFST of a lexicon.

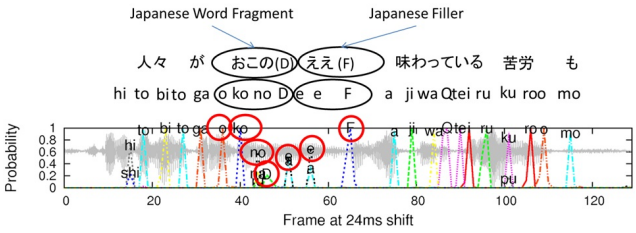
2. ACOUSTIC MODEL

In this section, we introduce an acoustic model developed in [7]. In order to model and discriminate phonetic units and acoustic events simultaneously, we used Long Short-Term Memory trained by Connectionist Temporal Classification (LSTM-CTC) criterion [13]. In an end-to-end approach [14], grammatical events such as "apostrophe" and "space" symbols are used as an output symbols of the model in addition to graphemes. On the other hand, in our approach, a filler symbol and a word fragment symbol are added to the output symbols of the model. Since LSTM can model long-term dependency between input and output sequences and CTC allows to train a model using input and output sequences of different lengths without alignments, we only need to insert filler symbols and word fragment symbols at appropriate positions in output sequences to model these acoustic events. Table 1 shows an example of output sequences with a filler and a word fragment. In this table, <F> and <D> represent filler and word fragment symbols, respectively.

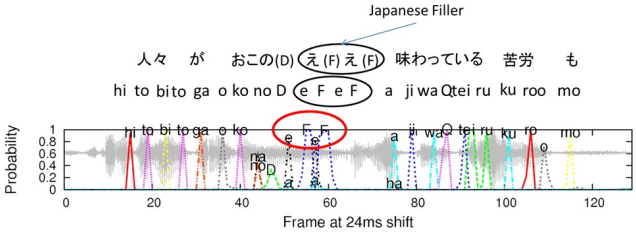
Figure 1 shows output probability sequences for Japanese morae and filler and word fragment symbols calculated by LSTM-CTC acoustic models. Figure 1(a) shows an output probability sequence using a label in which acoustic event symbols were inserted at the end of fillers and word fragments as described in [7]. On the other hand, in Figure 1(b), filler symbols are inserted after each mora in fillers. In this case, it is possible to calculate confidence scores for filler detection by counting the number of filler symbols in word hypotheses as described in the next section. It is noted that the con-

*Equal Contribution

†Currently with Toshiba Digital Solutions Corporation.



(a) Acoustic event symbols are inserted at the end of fillers and word fragments.



(b) Filler symbols are inserted after each mora in fillers.

Fig. 1. Probability sequences of phonetic and acoustic event symbols

Table 1. Output sequences with fillers and word fragments

Transcription	Thi-this is ah a pen
Phonetic symbols	di di s Iz aa a pe N
+filler and word fragment symbols	di <D> di s Iz aa <F> a pe N

fidence scores cannot be calculated for word fragments because it is thought that initial parts of word fragments do not have distinct features compared to that of normal words, and thus we insert word fragment symbols only at the end of word fragments.

3. DECODER

We adopt a WFST decoder with an LSTM-CTC acoustic model [11]. The decoder uses a WFST composed of R, L and G as a decoding graph, where R is a WFST squashing a label sequence of the acoustic model in a CTC manner, L is a WFST of a lexicon and G is a WFST of a language model. In the proposed method, a filler symbol <F> does not appear in L unlike [7] in order to recognize arbitrary word as filler. For recognizing word fragments, The decoder should accept arbitrary phonetic symbol sequences terminated by a word fragment symbol <D>. This can be achieved by extending a mechanism of dynamic words which are not in L [12, 15, 16].

3.1. Filler

In order to detect fillers, transitions accepting the filler symbol are added to the WFST R. The filler symbol <F> appears only on the input side of the transitions as well as a blank. By checking input symbols on the transitions of word hypotheses, it is possible to detect words as fillers. Figure 2 shows an example of the WFST R.

Here we introduce a filler confidence score $c = f/p$, where f and p are the numbers of detected filler symbols and phonetic symbols in a word, respectively. The decoder outputs a word as a filler only when c is higher than a predetermined threshold.

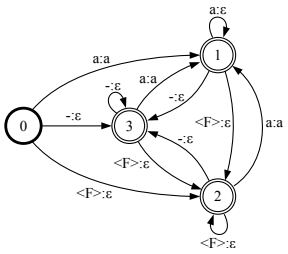


Fig. 2. WFST R with transitions accepting a filler symbol <F>.

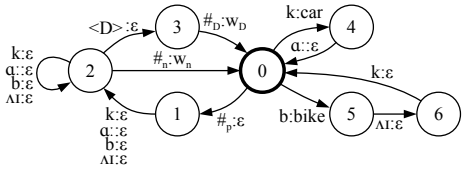


Fig. 3. WFST L with phoneme loops.

3.2. Word Fragment

In order to detect word fragments without registering them to the lexicon, a WFST L should accept arbitrary phonetic symbol sequences terminated by <D>. Figure 3 shows an example of L including transitions to recognize arbitrary phonetic symbol sequences. In this example, ε is an empty symbol, a set of phonetic symbols is P ={k, b, ɑː, ʌɪ}, and a set of normal words is {car, bike}. $\#_{\mathrm{D}}$, $\#_{\mathrm{n}}$ and $\#_{\mathrm{p}}$ are auxiliary symbols to help determinization. w_{D} is a symbol to help determining where the word fragments appear in G. w_{n} has the same role as w_{D} but for dynamic words. State 2 has self-transitions, each of which has a phonetic symbol in P on the input side for accepting arbitrary phonetic symbol sequences. Word fragments and dynamic words share these self-transitions.

In order to calculate probabilities of dynamic words and word fragments, the WFST G has to handle w_{n} and w_{D}. In the WFST G, w_{n} is treated as a normal word, and the probability of w_{n} is calculated in the same manner as normal words. This can be achieved by assigning probability of a class of words, such as unknown words or words with the same part-of-speech, to w_{n}. On the other hand, it is difficult to calculate probabilities of the word fragments because of the difficulty of collecting a text corpus including word fragments occurring naturally. Thus, G should accept w_{D} in any context by adding a self-transition with w_{D} to states in G. To prevent excessive enlargement of $L \circ G$, we limit adding the self-transition with w_{D} to states having an outgoing transition with w_{n}.

The construction steps for L and G is modified as

$$RLG = \pi_{\varepsilon}(opt(R \circ opt(\underset{i \to o}{proj}(L \circ G)))),$$

where $\circ$ represents composition, *opt* represents determinization and minimization, π_{ε} replaces auxiliary symbols with ε [17], and $proj_{i \to o}$ projects an input symbol to an output symbol for each transition in the cyclic transition of L for phonemes. Additionally, an output symbol on a transition with input symbol $\#_{\mathrm{p}}$ is replaced by w_{n} for dynamic words and w_{D} for word fragments, respectively, after $proj_{i \to o}$ for decoding efficiency.

Word fragments are recognized through a WFST D by sharing the mechanism of dynamic words without using outputs of RLG directly. The decoder refers $RLGD = RLG \circ D$ during decoding. This composition operation is performed on-the-fly. The WFST D has three functions: (1) convert a phoneme sequences to dynamic

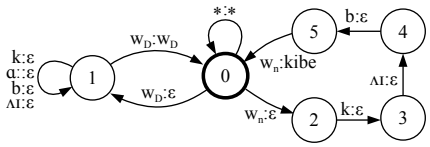


Fig. 4. WFST D.

words, (2) transfer normal words without any conversion, (3) convert phonetic symbol sequences to w_D which is referred in post-processing. Figure 4 shows an example of the WFST D. In this example, a dynamic word 'kibe' is recognized by a path $0 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 0$, a self-transition with *:* at state 0 realizes the function (2), and a path $0 \rightarrow 1 \rightarrow 0$ realizes the function (3).

The post-processing collects phonetic symbol sequences of word fragments by the following steps:

1. Find w_D on the output side of a searched path.
2. Find word boundaries of w_D.
3. Retrieve an input symbol sequence on the path specified by the word boundaries.
4. Squash the input symbol sequence in the CTC manner.
5. Get a character sequence from the squashed input symbol sequence such as a phonetic symbol sequence.

In the case of recognizing Japanese language using morae, a squashed input symbol sequence can be easily converted to a kana character sequence because they can be mapped one-to-one.

In order to prevent word fragments from dominating recognized words, a penalty is added to the self-transition for word fragments (e.g., at state 1 in Figure 4). This penalty should be determined empirically as described in the next section.

4. EXPERIMENTS

4.1. Experimental Setup

4.1.1. Evaluation Data

Evaluation experiments were conducted using an in-house Japanese corpus and the Corpus of Spontaneous Japanese (CSJ) [18]. The in-house Japanese corpus is a liaison-meeting evaluation set that consists of half an hour of speech by a speaker. The CSJ is a standard set for an evaluation of spontaneous Japanese speech recognition. We used CSJ testset3 that includes 2,484 monologue utterances by 10 speakers. For both of the evaluation sets, the utterances were recorded by close-talking microphones. The number of fillers and the number of word fragments in each evaluation set are shown in Table 2. The evaluation metrics are precision and recall rates for detection performance, and character error rate (CER) [%] for speech recognition performance, which is calculated by $CER = 100 - 100 \times (C - I)/N$, where C is the number of correct characters, I is the number of insertion characters, and N is the total number of characters in the reference.

4.1.2. Acoustic Model

In the experiments, uni-directional LSTM-CTC models were trained on the complete CSJ training set (about 580 hours) for real-time decoding. The CSJ training set has filler and word fragment tags in transcriptions. The basic feature was a 28-dimensional Mel-filterbank output. The input feature vector for the LSTM-CTCs was created by concatenating current and consecutive 8 previous frames, giving 9 frames in each input feature vector. Hidden layers were composed of 3 LSTM layers with 1024-dimensional memory cells and recurrent projections [19] that reduce the number of dimensions from 1024 to 256 for each layer output. A final layer of 126 units was set for Japanese mora outputs including a blank and a silence outputs. For filler and word fragment outputs, 2 units were added to the final layer. For all experiments, we created three acoustic models listed in Table 3. The main differences among them were the training labels. For fillers, two training methods were tested as shown in Figure 1. One was to insert filler symbols at the end of fillers as shown in Figure 1 (a), and the other was to insert filler symbols after each mora in fillers as shown in Figure 1 (b).

Table 2. *The number of fillers and word fragments*

	#Filler	#Word fragment
CSJ testset3	785	169
Liaison-meeting	238	36

Table 3. Acoustic models

Acoustic model	Training label for "こ, ええと この (ko, eeto kono)"
Normal model (NAM)	ko e e to ko no
Filler + word fragment detection model (FDAM1)	ko <D> e e to <F> ko no
Filler + word fragment detection model (FDAM2)	ko <D> e <F> e <F> to <F> ko no

Table 4. Lexicons for the conventional decoder

Lexicon	Word and the pronunciation
Normal lexicon (NLex)	kono: ko no, eeto: e e to
Normal lexicon with word fragments (NLex+D)	kono: ko no, ko: ko, eeto: e e to
Lexicon with filler and word fragment symbols 1 (FDLex1)	kono: ko no, ko: ko <D>, eeto: e e to <F>
Lexicon with filler and word fragment symbols 2 (FDLex2)	kono: ko no, ko: ko <D>, eeto: e <F> e <F> to <F>

4.1.3. Language Model and Lexicon

A 4-gram language model was trained on 200 million sentences extracted from web pages and Toshiba internal spontaneous dataset. It is noted that the CSJ dataset was not include in the training set.

Table 4 shows four types of lexicons and sample words included in the lexicons. In this table, the normal lexicon was used for the proposed decoder, and other lexicons are used for the conventional decoder. The vocabulary size was about 200,000 words. The lexicons included fillers extracted from the spontaneous dataset. Word fragments extracted from the CSJ dataset were added to the lexicons for the conventional decoder, while word fragments were not added to the normal lexicon for the proposed decoder. In order to combine LSTM-CTC acoustic models having filler and word fragment outputs with conventional decoder, filler and word fragment symbols were added to pronunciations of words in the lexicons FDLex1 and FDLex2.

4.1.4. WFST

We prepared 5 WFSTs based on the combinations of acoustic models and lexicons. For each WFST, we selected a conventional

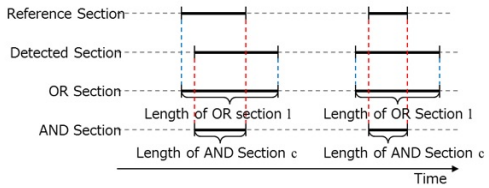


Fig. 5. The metric for calculating the tolerance

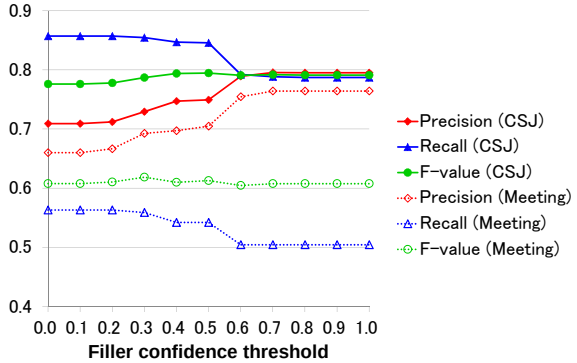


Fig. 6. Filler confidence and detection performance using FDWT4

decoder or the proposed decoder for detection of fillers and word fragments. Table 5 shows the combinations and selected methods. "NWT" is the WFST using the conventional language model, the conventional lexicon with word fragments, and the conventional decoder. "FDWT1" and "FDWT2" use lexicons including filler and word fragment symbols for the decoding. The main contributions of this paper are "FDWT3" and "FDWT4" that use the conventional language model, the conventional lexicon and the proposed decoder. We used a Toshiba original decoder based on [15] for all experiments.

4.2. Speech Recognition Performance

Table 6 shows the recognition performance of each WFST. Basically CERs are similar among the WFSTs. However, the CERs of FDWT1-4 are slightly better than NWT's. It seems that the explicit modeling of fillers and word fragments results in better performance.

4.3. Detection Performance

We defined a tolerance for filler and word fragment detection based on logistic OR and logistic AND sections between reference and detected sections depicted in Figure 5. The tolerance is calculated by $tolerance = (l - c)/c$, where l and c denote the length of the OR section and the length of the AND section, respectively. The value becomes 0 when an detected section is identical to a corresponding reference section. In this experiment, detection of fillers and word fragments were judged to be correct if the tolerance was less than a threshold value. We set the threshold value to 0.7 for filler detection and 0.9 for word fragment detection. Generally, uni-directional LSTM-CTC modeling leads to the delay of output of phonetic symbols. Hence, the positions of the detections are uniformally shifted back compared to the real positions of the acoustic events on the time axis. We set the time offset to 300 ms to compensate for the difference.

Figure 6 shows the relation between the filler confidence threshold and the filler detection performance in terms of the precision and recall rates and the F-value. Table 7 and 8 show the performance of the filler and word fragment detection of each WFST. The value of filler confidence was set to 0.3 in Table 7. The penalty to the self-transition for word fragments was set to maximize the ASR performance. For the filler detection, it can be shown that the performance of FDWT1-4 was better than that of NWT. However, the performance of FDWT3 and FDWT4 was similar to that of FDWT1 and FDWT2. This is because the filler can likely be covered with a conventional language model and lexicon (NWT), and the effectiveness of the acoustic assist for the filler detection is the same for all methods. The advantage of FDWT4 is that it can control the precision and recall rates by the filler confidence score depending on the purposes of applications. On the other hand, for word fragment detection, the performance of FDWT3 and FDWT4 was apparently better than the performance of FDWT1 and FDWT2. The conventional decoder cannot detect word fragments even if word fragments extracted from the CSJ dataset is added to the lexicon. These results show that our approach is effective for detection of filler and word fragment using conventional language models and lexicons.

Table 5. Created WFSTs ("WF": Word Fragment)

WFST name	Acoustic model	Lexicon	Decoder
Normal (NWT)	NAM	NLex+D	Conventional
Filler & WF (FDWT1)	FDAM1	FDLex1	Conventional
Filler & WF (FDWT2)	FDAM2	FDLex2	Conventional
Filler & WF (FDWT3)	FDAM1	NLex	Proposed
Filler & WF (FDWT4)	FDAM2	NLex	Proposed

Table 6. ASR performance for each WFST (CER [%])

WFST name	CSJ testset3	Liaison-meeting	Ave.
NWT	10.34	15.36	12.85
FDWT1	10.35	14.75	12.55
FDWT2	9.83	14.65	12.24
FDWT3	10.53	14.82	12.68
FDWT4	10.16	14.99	12.57

Table 7. Filler Detection performance (F:F-value)

WFST	CSJ testset3			Liaison-meeting		
	Precision	Recall	F	Precision	Recal	F
NWT	0.77	0.61	0.68	0.79	0.45	0.57
FDWT1	0.80	0.77	0.78	0.87	0.57	0.69
FDWT2	0.78	0.81	0.79	0.78	0.58	0.66
FDWT3	0.75	0.85	0.79	0.72	0.58	0.64
FDWT4	0.75	0.85	0.79	0.70	0.54	0.61

Table 8. Word Fragment Detection performance (F:F-value)

WFST	CSJ testset3			Liaison-meeting		
	Precision	Recall	F	Precision	Recall	F
NWT	0.22	0.04	0.07	0.00	0.00	0.00
FDWT1	0.36	0.02	0.04	1.00	0.03	0.05
FDWT2	0.42	0.03	0.06	1.00	0.06	0.11
FDWT3	0.53	0.25	0.34	0.86	0.33	0.48
FDWT4	0.49	0.25	0.34	0.61	0.31	0.41

5. CONCLUSION

We proposed a technique to detect fillers and word fragments using an LSTM acoustic model and a WFST decoder. The LSTM acoustic model outputs filler and word fragment symbols as well as phonetic symbols, and the proposed decoder can simultaneously recognize speech and detect fillers and word fragments using conventional language models and lexicons without registering all possible word fragments. We showed that in word fragment detection the proposed decoder outperformed a conventional decoder using lexicons including word fragments.

6. REFERENCES

[1] Takahiro Shinozaki, Chiori Hori, and Sadaoki Furui, "Towards automatic transcription of spontaneous presentations," in *Eurospeech*, 2001, vol. 1, pp. 491–494.

[2] Ariya Rastrow, Abhinav Sethy, and Bhuvana Ramabhadran, "A new method for oov detection using hybrid word/fragment system," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*. IEEE, 2009, pp. 3953–3956.

[3] Hong-Kwang Kuo, Ellen Eide Kislal, Lidia Mangu, Hagen Soltau, and Tomas Beran, "Out-of-vocabulary word detection in a speech-to-speech translation system," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 7108–7112.

[4] Taichi Asami, Ryo Masumura, Yushi Aono, and Koichi Shinoda, "Recurrent out-of-vocabulary word detection using distribution of features.," in *INTERSPEECH*, 2016, pp. 1320–1324.

[5] Keikichi Hirose, Yu Abe, and Nobuaki Minematsu, "Detection of fillers using prosodic features in spontaneous speech recognition of japanese," in *Proceedings of International Conference on Speech Prosody, Dresden*, 2006, vol. 2, pp. 2–5.

[6] Yulia Tsvetkov, Zaid Sheikh, and Florian Metze, "Identification and modeling of word fragments in spontaneous speech," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 7624–7628.

[7] Yu Nasu and Hiroshi Fujimura, "Acoustic event detection and removal using LSTM-CTC for speech recognition (in Japanese)," in *IEICE Tech. Rep.,PRMU2016-69*, 2016, vol. IEICE-116, pp. 121–126.

[8] Dzmitry Bahdanau, Jan Chorowski, Dmitriy Serdyuk, Philemon Brakel, and Yoshua Bengio, "End-to-end attention-based large vocabulary speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 4945–4949.

[9] Suyoun Kim, Takaaki Hori, and Shinji Watanabe, "Joint ctc-attention based end-to-end speech recognition using multitask learning," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 4835–4839.

[10] Rohit Prabhavalkar, Kanishka Rao, Tara N Sainath, Bo Li, Leif Johnson, and Navdeep Jaitly, "A comparison of sequence-to-sequence models for speech recognition," *Proc. Interspeech 2017*, pp. 939–943, 2017.

[11] Yajie Miao, Mohammad Gowayyed, and Florian Metze, "EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding," in *Proceedings of the ASRU 2015*. IEEE, 2015, pp. 167–174.

[12] Paul R. Dixon, Chiori Hori, and Hideki Kashioka, "A specialized WFST approach for class models and dynamic vocabulary," in *Proceedings of the INTERSPEECH 2012*. ISCA, 2012.

[13] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *Proc. ICML*, 2006, pp. 369–376.

[14] Dario Amodei, Rishita Anubhai, Eric Battenberg, Carl Case, Jared Casper, Bryan C. Catanzaro, Jingdong Chen, Mike Chrzanowski, Adam Coates, Greg Diamos, Erich Elsen, Jesse Engel, Linxi Fan, Christopher Fougner, Tony Han, Awni Y. Hannun, Billy Jun, Patrick LeGresley, Libby Lin, Sharan Narang, Andrew Y. Ng, Sherjil Ozair, Ryan Prenger, Jonathan Raiman, Sanjeev Satheesh, David Seetapun, Shubho Sengupta, Yi Wang, Zhiqian Wang, Chong Wang, Bo Xiao, Dani Yogatama, Jun Zhan, and Zhenyao Zhu, "Deep Speech 2: End-to-end speech recognition in English and Mandarin," arXiv:1512.02595, 2015.

[15] Manabu Nagao, "Generation device, recognition device, generation method, and computer program product," U. S. Patent Publication No. 2016/0155440.

[16] Cyril Allauzen and Michael Riley, "Rapid vocabulary addition to context-dependent decoder graphs," in *Proceedings of the INTERSPEECH 2015*. ISCA, 2015, pp. 2112–2116.

[17] Mehryar Mohri, Fernando Pereira, and Michael Riley, "Weighted finite-state transducers in speech recognition," in *ASR-2000*, 2000, pp. 97–106.

[18] Tatsuya Kawahara, Hiroaki Nanjo, Takahiro Shinozaki, and Sadaoki Furui, "Benchmark test for speech recognition using the corpus of spontaneous japanese," in *ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition*, 2003.

[19] Haşim Sak, Andrew Senior, and Françoise Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *Proc. Fifteenth Annual Conference of the International Speech Communication Association (INTERSPEECH2014)*. ISCA, 2014, pp. 338–342.