ACOUSTIC-TO-WORD ATTENTION-BASED MODEL COMPLEMENTED WITH CHARACTER-LEVEL CTC-BASED MODEL

Sei Ueno, Hirofumi Inaguma, Masato Mimura, Tatsuya Kawahara

Graduate School of Informatics, Kyoto University, Sakyo-ku, Kyoto, Japan

ABSTRACT

This paper addresses end-to-end speech recognition which directly maps acoustic features to a word sequence. The acoustic-to-word model is attractive since it does not require an external language model and an elaborate decoder, resulting in extremely simple and fast decoding. The apparent drawback of this modeling is sparseness of training data, particularly for less frequent words. In this paper, we propose a framework complemented with a character-level model. Joint training of the word-level model with the character-level model enhances the generality of deep learning of feature extraction and classification processes, preventing it from overfitting. Moreover, the character-level model is used to decode out-of-vocabulary (OOV) words that are not covered by the word-level model. Since there are choices of connectionist temporal classification (CTC) and attention-based models in the end-to-end recognition, we also explore optimal combination for the hybrid system. Evaluations on the Corpus of Spontaneous Japanese (CSJ) show that (1) the acoustic-to-word attention-based model outperforms CTC, (2) multitask learning (MTL) with character-level CTC model is effective, and (3) the hybrid system achieves comparable or even better accuracy than the standard DNN-HMM system with a decoding speed faster by a factor of 25.

Index Terms— End-to-end speech recogntion, acoustic-to-word, attention, connectionist temporal classification (CTC), multitask learning

1. INTRODUCTION

Deep neural networks (DNNs) have drastically improved the performance of automatic speech recognition (ASR). It was recently reported that even human-parity recognition performance can be achievable for the conversational telephone speech task [1, 2]. However, in exchange for the excellent performance, these ASR systems have very complicated architectures consisting of complex decoders, large language models, and carefully designed pronunciation dictionaries. They have a large runtime latency and less portability.

On the other hand, a much simpler architecture of end-to-end speech recognition has been investigated intensively. It is formulated to map input acoustic features into a target symbol sequence with recurrent neural networks (RNN) such as LSTMs, without requiring latent state transition models such as HMMs. There are two major approaches: one is connectionist temporal classification (CTC) [3, 4, 5, 6] that marginalizes and condenses all possible frame-wise output symbol sequences, and the other is the encoder-decoder model with an attention mechanism [7, 8, 9, 10, 11], which first encodes the input into a frame-wise distributed representation with one LSTM and then decodes it to a target symbol sequence with another LSTM. However, the conventional end-to-end systems are still based on subword units, such as phones, syllables and characters, and they still need a pronunciation lexicon and language model for transducing audio features into a word sequence [12].

Most recently, several studies [13, 14, 15] investigated end-to-end speech recognition using whole words as acoustic units. As a language model is also implicitly embedded in the RNN, this acoustic-to-word modeling does not require an external language model. It will realize an extremely simple and fast decoding only with the RNN. In this paper, we first investigate this modeling with a Japanese large-vocabulary ASR corpus, comparing the CTC and attention-based models.

The apparent problem of this approach is sparseness of training data, since the number of word entries is much larger than that of the subword units, and the distribution of the occurrence counts of words is much more unbalanced than that of subword units. Most critically, there are many entries of infrequent words which cannot be provided with a sufficient amount of training data. If we eliminate infrequent words for reliable model training, these words cannot be decoded in theory.

To address this problem, in this paper, we propose a framework complemented with a character-level model. This is regarded as a hybrid system of a word-level model and a character-level model, which share the network partially. Multitask learning (MTL) of these models is implemented to improve the generality of the network and prevent overfitting of the models of less frequent words. We also investigate possible choices of the auxiliary model including attention-based model and CTC. The auxiliary character-level model is also useful for the decoding stage to recover out-of-vocabulary (OOV) words, which are actually eliminated due to their infrequent occurrence. When the word-level model generates a segment of OOV words, the proposed hybrid system falls back to the character-level model to get an aligned sequence.

In the rest of the paper, we first review the modeling for end-to-end speech recognition in Section 2 and the basic concept of acoustic-to-word model in Section 3. Then, Section 4 gives explanations of the proposed hybrid framework including the multitask learning and parallel decoding. Experimental evaluations using the Corpus of the Spontaneous Japanese (CSJ) and Japanese Newspaper Article Sentences (JNAS) are presented in Section 5.

2. MODELS FOR END-TO-END SPEECH RECOGNITION

We first review two basic approaches to end-to-end speech recognition. Let $\boldsymbol{X} = (\boldsymbol{x}_1, ..., \boldsymbol{x}_T)$ denote a length-T sequence of input acoustic features. Let $\boldsymbol{y} = (y_1, ..., y_L)$ denote a length-L sequence of target labels which usually consist of phones or characters, where $y_l \in \{1, ..., K\}$ and K is the number of target labels.

978-1-5386-4658-8/18/$31.00 ©2018 IEEE

2.1. Connectionist Temporal Classification (CTC)

The main idea of CTC is to allow an extra "blank" label ϕ in order to learn a mapping between sequences of different lengths. By inserting a blank label between two consecutive labels and allowing each label to be repeated, a label sequence $\boldsymbol{y}$ can be expanded to a set of length-T sequences $\Omega(\boldsymbol{y})$. Inversely, each CTC path $\boldsymbol{\pi} \in \Omega(\boldsymbol{y})$ can be reduced to the original label sequence $\boldsymbol{y}$ after removing all repeating labels and blank labels, where $\boldsymbol{\pi} = (\pi_1, ..., \pi_T)$ and $\pi_t \in \{1, ..., K\} \cup \{\phi\}$.

The CTC loss is defined using the probabilities of all CTC paths included in $\Omega(\boldsymbol{y})$ as:

$$p(\boldsymbol{y}|\boldsymbol{X}) = \sum_{\pi \in \Omega(\boldsymbol{y})} p(\pi|\boldsymbol{X}) = \sum_{\pi \in \Omega(\boldsymbol{y})} \prod_{t=1}^{T} p(\pi_t|\boldsymbol{x}_t) \qquad (1)$$

where the posterior probabilities $p(\pi_t|\boldsymbol{x}_t)$ are calculated with a multi-layer bidirectional RNN.

The CTC loss and its gradient with respect to the network parameters are efficiently computed with the forward-backward algorithm. CTC-based models do not explicitly learn the internal relationship between labels since they assume that the probability of each label is independent of others.

2.2. Attention-based encoder-decoder model

An alternative approach to end-to-end mapping between speech and label sequences is to use the attention-based encoder-decoder architecture [7, 8, 9, 10, 11]. This architecture has two distinct subnetworks. One is the encoder subnetwork, which transforms an acoustic feature sequence to a sequential representation of length T. Based on this encoded information, the decoder subnetwork predicts a label sequence whose length, L, is usually shorter than the input. The decoder uses only a relevant portion of the encoded sequential representations for predicting a label at each time step using the attention mechanism.

The encoder is implemented as a multi-layer bidirectional RNN such as an LSTM, and the decoder usually consists of a 1-layer of unidirectional RNN followed by a softmax output layer.

The attention-based model is formulated as follows. The encoder transforms $\boldsymbol{X}$ to intermediate representation vectors $\boldsymbol{H} = (\boldsymbol{h}_1, ..., \boldsymbol{h}_T)$. In the following decoding step, the hidden state (memory) activation of the RNN-based decoder at the l-th time step is computed as:

$$\boldsymbol{s}_l = Recurrency\left(\boldsymbol{s}_{l-1}, \boldsymbol{g}_l, \boldsymbol{y}_{l-1}\right) \qquad (2)$$

where $\boldsymbol{g}_l$ and $\boldsymbol{y}_{l-1}$ denote the "glimpse" at the l-th time step and the predicted label at the previous step. The glimpse $\boldsymbol{g}_l$ is a weighted sum of the encoder output sequence as:

$$\boldsymbol{g}_l = \sum_t \alpha_{l,t} \boldsymbol{h}_t \qquad (3)$$

where $\alpha_{l,t}$ is an attention weight of $\boldsymbol{h}_t$. It is calculated as:

$$\begin{aligned} e_{l,t} &= Score(\boldsymbol{s}_{l-1}, \boldsymbol{h}_t, \boldsymbol{\alpha}_{l-1}) & (4) \\ \alpha_{l,t} &= \exp(e_{l,t}) / \sum_{t'=1}^{T} \exp(e_{l,t'}) & (5) \end{aligned}$$

There are many choices for implementation of the score function (4). In this paper, we adopt the hybrid location and content-based attention mechanism as follows:

$$\begin{aligned} e_{l,t} &= \boldsymbol{w}^T tanh(\boldsymbol{W}\boldsymbol{s}_{l-1} + \boldsymbol{V}\boldsymbol{h}_t + \boldsymbol{U}f_{l,t} + \boldsymbol{b}) & (6) \\ \boldsymbol{f}_l &= \boldsymbol{F} * \boldsymbol{\alpha}_{l-1} & (7) \end{aligned}$$

where $*$ denotes 1-dimensional convolution. Using $\boldsymbol{g}_l$ and $\boldsymbol{s}_{l-1}$, the decoder predicts the next label $\boldsymbol{y}_l$ as:

$$\boldsymbol{y}_l \sim Generate\left(\boldsymbol{s}_{l-1}, \boldsymbol{g}_l\right) \qquad (8)$$

where the Generate function is implemented as:

$$\boldsymbol{R} \tanh\left(\boldsymbol{P}\boldsymbol{s}_{l-1} + \boldsymbol{Q}\boldsymbol{g}_l\right) \qquad (9)$$

The objective function for training the attention models is a cross entropy loss calculated between the predicted label sequences and the target correct label sequences. In end-to-end speech recognition using the attention model, we use special labels for denoting start-of-sentence (sos) and end-of-sentence (eos). The decoder completes decoding an utterance when the end-of-sentence is emitted. It is possible to conduct a beam search to further enhance the recognition performance.

3. ACOUSTIC-TO-WORD MODELING

Most recently, word-level end-to-end speech recognition is investigated by exploiting bidirectional LSTM-based models [13, 14, 15]. The remarkable advantages of this acoustic-to-word modeling are its decoding speed and drastically simplified architecture. It does not require external decoders, language models, and a pronunciation dictionary. Speech recognition can be conducted simply by picking up the output of the neural network.

In this acoustic-to-word modeling, we have a choice of CTC and attention-based models. While Lu et al. [15] investigated encoder-decoder models, Soltau et al. [14] and Audhkhasi et al. [13] presented good results with CTC-based models. In theory, an attention-based model explicitly incorporates contextual information from the target label sequence in the decoder RNN, while the CTC model only considers frame-level contexts in the encoder LSTM. Since the word unit is usually longer than the subword unit, the frame-level contextual model may not be sufficient and thus the attention-based model is expected to work better. In this paper, we first investigate the attention-based model for acoustic-to-word modeling in comparison with the CTC-based model.

On the other hand, the acoustic-to-word modeling has an apparently serious drawback in training. Compared with the conventional subword-based modeling, the number of word entries is much larger and the distribution of their occurrence counts is much more unbalanced. Therefore, it is expected that many words will not have sufficient training data, leading to overfitting. Audhkhasi et al. [13] reported that acoustic-to-word models initialized with random values did not converge when the amount of training data was limited. Thus, this modeling requires a huge amount of training data, which may not be available in many languages and application domains.

Another problem inherent to the word-based model is that it can only decode words covered in the training stage. It is not possible to add new words to the dictionary by providing their baseforms as with subword unit modeling. Furthermore, as addressed above, many word entries with a lower occurrence count need to be eliminated due to insufficient training data, thus they cannot be recognized, either.

To solve these problems, in this paper, we propose to complement acoustic-to-word modeling with character-level modeling.

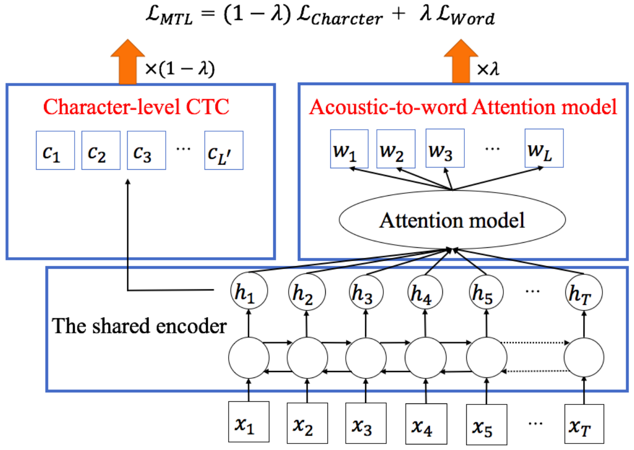
$\mathcal{L}_{MTL} = (1-\lambda)\,\mathcal{L}_{Charcter} + \lambda\,\mathcal{L}_{Word}$
×(1 − λ)
×λ
Character-level CTC
c_1 c_2 c_3 ⋯ $c_{L'}$
Acoustic-to-word Attention model
w_1 w_2 w_3 ⋯ w_L
Attention model
The shared encoder
h_1 h_2 h_3 h_4 h_5 ⋯ h_T
x_1 x_2 x_3 x_4 x_5 ⋯ x_T

Fig. 1. *MTL loss is a weighted sum of the CTC loss and the loss of the attention model.*

4. WORD ATTENTION MODEL COMPLEMENTED WITH CHARACTER CTC MODEL

4.1. Multitask learning with character-level CTC loss

In order to mitigate the problem in the training of the word-level model, it is straightforward to use a subword-level model as a basis or initialization. Audhkhasi et al. [13] pre-trained the lower part of a word-level model using another CTC loss criterion with subwords as acoustic units. It is also reasonable to use subword-level supervision for training a word-level model as Toshniwal et al. used senone-level supervision for subword-level training [16]. Meanwhile, Kim et al. [17] showed that an attention-based model could be improved by enforcing monotonic alignment between speech and label sequences. This was done by adding another output layer after the encoder with an auxiliary task with a CTC loss.

Taking these findings into account, we propose multitask learning (MTL), in which the word-level attention model is complemented with a character-level CTC model. We expect that the joint training with the character-level model alleviates the problem of data sparseness in less frequent words and improves the generality of the model. Moreover, using a CTC loss in this auxiliary task will enforce the monotonic constraint during training

Fig. 1 illustrates the overall architecture of our proposed framework. In the encoder step, the entire model uses the shared network. Then, there are two branches in the decoding step. One is the word-level attention-based decoder with the number of output nodes equal to the vocabulary size. The other branch corresponds to the character-level model that functions as an auxiliary task and propagates a CTC loss. The objective function for the MTL is defined as a weighted sum of the losses propagated from both branches.

4.2. Resolving OOV words

Another serious drawback of the acoustic-to-word model is that it cannot decode an OOV word. As a remedy for this problem, we can also turn to the character-level model.

When an <UNK> symbol, which models OOV words, is predicted by the word-level model, the speech segment of the OOV word can be roughly spotted by picking up the frame that has the largest value in the attention vector. Meanwhile, segmentation of the CTC output can be identified. In the character-level model, we have a special character for denoting word boundaries (<wb>). Thus, we can align the segment of the <UNK> model with the sequence between the two word boundary symbols, and output the corresponding character sequences. Fig. 2 illustrates an example of resolving an OOV word by this procedure. The time frame of the maximum value in the attention vector when producing the <UNK> symbol corresponds to the segment of *E* in the character-level model, thus the sequence which includes *E* between <wb>, that is *DEF*, is regarded as the resolved sequence of the OOV word.

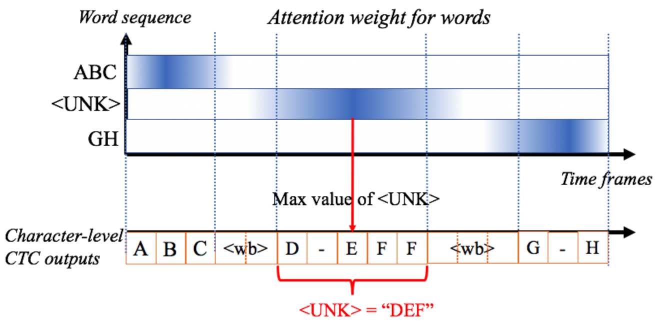
Word sequence
Attention weight for words
ABC
<UNK>
GH
Time frames
Max value of <UNK>
Character-level CTC outputs
A B C <wb> D - E F F <wb> G - H
<UNK> = "DEF"

Fig. 2. *An example of resolving an OOV word. When the word-level model outputs UNK, the time frame of the maximum value in the attention vector is identified. In the character-level model, the corresponding sequence of the characters between the word boundaries (wb) is extracted for output.*

We expect that this mechanism decodes some OOV words and contributes to reduced WER. Even if only a part of an OOV word is recovered, the output would be more informative for users than an <UNK> symbol.

5. EXPERIMENTAL EVALUATIONS

5.1. Data and task

We evaluated our methods through three speech recognition tasks using two standard Japanese corpora: the Corpus of Spontaneous Japanese (CSJ) [18] and Japanese Newspaper Article Sentences (JNAS) [19]. CSJ includes two distinct sub-corpora, namely, CSJ-APS and CSJ-SPS. CSJ-APS consists of academic public speeches on several topics such as science, engineering, humanities and social science. CSJ-SPS consists of simulated public speeches on three themes. JNAS is a Japanese read speech corpus of newspaper articles. In Japanese, different character sets, namely, Hiragana, Katakana, Kanji, and Roman alphabets are used in a mixed manner. Therefore, there are many more characters in Japanese than, for example, graphemes in English. Table 1 shows the number of distinct words and characters used for the models of each corpus.

The corpora we used for training models have their own official test sets. CSJ-TESTSET1 for CSJ-APS consists of 10 academic lectures by 10 male speakers. CSJ-TESTSET3 for CSJ-SPS consists of 10 simulated speeches by 5 male and 5 female speakers. The test set of JNAS consists of 200 sentences spoken by 23 male and female speakers. The OOV rates of CSJ-TESTSET1, CSJ-TESTSET3, and the JNAS test set are 1.36%, 1.48% and 2.20%, respectively.

5.2. System configuration

A 120-dimensional feature vector of 40-channel log Mel-scale filterbank (lmfb) outputs and their delta and acceleration coefficients are used as acoustic features. Non-overlapping frame stacking [5] was

Table 1. *The number of distinct words and characters in each training set. The word vocabulary includes words which appear more than three times. Others are treated as OOV words (UNK). The word vocabulary also includes three special symbols of pause, start of sentence (sos), and end of sentence (eos). In the character vocabulary, UNK is not included but word boundary (wb) is added.*

	CSJ-APS	CSJ-SPS	JNAS
# words	19146	24826	10612
# characters	2854	3039	2315

Table 2. *ASR performance (WER and real-time factor) of CSJ-APS testset (224 hours). (·) means the model for an auxiliary task.*

Model	WER(%)	RTF
DNN-HMM + largeLM	13.62	0.925
phone CTC + largeLM	14.15	0.581
word CTC	16.97	0.010
word attention	14.67	0.035
word attention (character CTC)	13.84	0.035
+ resolving unknown words	13.65	0.035

applied to these features, in which we stack and skip three frames to make a new super-frame.

The acoustic encoder in the attention-based model consists of three-layers of bidirectional LSTMs with 320 cells. The dropout [20] rate was 0.2 for training each BiLSTM layer. The attention-based word decoder consists of one-layer LSTM with 320 cells, a hidden layer with 320 tanh nodes, and a softmax output layer for word entries. The weight of the word-level model was set to be 0.8 and that for the character-level CTC was 0.2.

We used the Adam method with a standard setting described in [21] for optimizing networks. We also used gradient clipping with a threshold of 5.0. The minibatch size was set to be 30 through all experiments, except for a preliminary experiment for comparing different MTL strategies (Table 3). All network parameters were initialized with random values drawn from a uniform distribution with a range (-0.1, 0.1). Since providing long input sequences can slow convergence at the beginning of the training, the input data were sorted by the length of frames before creating minibatches. We used the Chainer toolkit [22] to train the networks. In decoding with the word-level attention model, we used a simple beam search with the beam width of 4.

We also built a DNN-HMM hybrid system and a phone-CTC system using CSJ-APS as baselines. The DNN-HMM system has seven hidden layers with 2k sigmoidal nodes and a softmax output layer with 5k nodes. It was trained using a sequence discriminative criterion. The phone CTC system has three layers of bidirectional LSTMs and a softmax output layer. In decoding with these baseline systems, we used a large language model trained using transcription of both of CSJ-APS and CSJ-SPS, utilizing their advantage that they can use external large language models. For decoding with the hybrid DNN-HMM and the phone CTC, we used the Julius decoder and the EESEN WFST decoder, respectively [12].

5.3. Results

Table 2 shows the ASR performance in WER as well as the real-time factor (RTF) for CSJ-TESTSET1. All models were trained using CSJ-APS. We can see that the acoustic-to-word model completes decoding in significantly shorter time than the baselines. The word attention model outperformed the CTC-based word model by 2.3 points in WER. This confirms that the attention-based model is advantageous because it can learn a word-level language model. The proposed MTL method yielded a further improvement of 0.83 points.

Table 3. *Comparison of auxiliary models for MTL on CSJ-APS. In this experiment, minibatch = 10. Acoustic-to-word attention model is used. "—-" row means acoustic-to-word attention model only.*

Model for auxiliary task	WER(%)
—-	17.25
character CTC (Proposed)	**16.11**
character attention	17.55
word CTC	17.03

Table 4. *ASR performance on different corpora (WER)*

	CSJ-SPS	CSJ-APS	JNAS
Training data amount	251hrs	224hrs	85hrs
DNN-HMM+LM	12.80	13.62	7.18
word attention	12.23	14.67	23.99
word attention (character CTC)	12.13	13.84	19.36
+ resolving unknown word	11.95	13.65	19.07

Although we confirmed that MTL using an auxiliary task of a character-level CTC loss is effective in Table 2, we also compared other auxiliary tasks. The results obtained with the word-level attention model combined with character-level attention model or word-level CTC model are shown in Table 3. The word CTC auxiliary task yielded a slight improvement of 0.22 points, but the character attention models did not improve the performance.

The OOV resolution method described in Section 4.2 was also performed. It reduced WER by 0.19 points, achieving a comparable performance to the baseline DNN-HMM system with more than 25-times faster decoding speed.

Table 4 compares the ASR performance for different corpora. It is observed that the performance of the word-level model is better when the amount of the training data is larger. The word attention model yielded even better performance for the CSJ-SPS test set than the DNN-HMM. On the other hand, in the JNAS test set, the word-level model showed only poor performance compared with the baseline DNN-HMM, showing that the word-level model needs a considerable amount of data. Another interesting point is that MTL is more effective when the training data is small or the task is difficult. We also confirmed that the OOV resolution method yielded small but consistent improvements for all test sets.

6. CONCLUSIONS

We have proposed joint training of a word-level model with a character-level model to solve the sparseness of training data, particularly for less frequent words. We first showed that the acoustic-to-word attention-based model outperformed CTC. The proposed method achieved comparable or even better accuracy than the DNN-HMM system with a significantly faster speed on two tasks. We also demonstrated that decoding unknown words using the character-level model is also effective for further improving the accuracy.

7. REFERENCES

[1] George Saon, Gakuto Kurata, Tom Sercu, Kartik Audhkhasi, Samuel Thomas, Dimitrios Dimitriadis, Xiaodong Cui, Bhuvana Ramabhadran, Michael Picheny, Lynn-li Lim, Bergul Roomi, and Phil Hall, "English Conversational Telephone Speech Recognition by Humans and Machines," in *Interspeech 2017*, 2017.

[2] Wayne Xiong, Jasha Droppo, Xuedong Huang, Frank Seide, Mike Seltzer, Andreas Stolcke, Dong Yu, and Geoffrey Zweig, "The Microsoft 2016 conversational speech recognition system," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, 2017, pp. 5255–5259.

[3] Alex Graves, Santiago Fernandez, Faustino Gomez, and Jurgen Schmidhuber, "Connectionist temporal classification : Labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine Learning*, 2006, pp. 369–376.

[4] Alex Graves and Navdeep Jaitly, "Towards End-To-End speech recognition with recurrent neural networks," in *Proceedings of the 31st International Conference on Machine Learning*, 2014, pp. 1764–1772.

[5] Haşim Sak, Andrew Senior, Kanishka Rao, and Françoise Beaufays, "Fast and accurate recurrent neural network acoustic models for speech recognition," *arXiv preprint arXiv:1507.06947*, 2015.

[6] Haşim Sak, Félix de Chaumont Quitry, Tara Sainath, and Kanishka Rao, "Acoustic modelling with CD-CTC-SMBR LSTM RNNs," in *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*. IEEE, 2015, pp. 604–609.

[7] Jan Chorowski, Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, "End-to-end continuous speech recognition using attention-based recurrent NN: first results," in *NIPS: Workshop Deep Learning and Representation Learning Workshop*, 2014.

[8] Jan Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio, "Attention-based models for speech recognition," in *Advances in Neural Information Processing Systems (NIPS)*, 2015, pp. 577–585.

[9] Rohit Prabhavalkar, Tara N Sainath, Bo Li, Kanishka Rao, and Navdeep Jaitly, "An analysis of "attention" in sequence-to-sequence models," in *Interspeech 2017*, 2017, pp. 3702–3706.

[10] William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, 2016, pp. 4960–4964.

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

[12] Yajie Miao, Mohammad Gowayyed, and Florian Metze, "EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding," in *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*, 2015, pp. 167–174.

[13] Kartik Audhkhasi, Bhuvana Ramabhadran, George Saon, Michael Picheny, and David Nahamoo, "Direct acoustics-to-word models for English conversational speech recognition," in *Interspeech2017*, 2017, pp. 959–963.

[14] Hagen Soltau, Hank Liao, and Hasim Sak, "Neural speech recognizer: Acoustic-to-word LSTM model for large vocabulary speech recognition," in *Interspeech2017*, 2017, pp. 3707–3711.

[15] Liang Lu, Xingxing Zhang, and Steve Renals, "On training the recurrent neural network encoder-decoder for large vocabulary end-to-end speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, 2016, pp. 5060–5064.

[16] Shubham Toshniwal, Hao Tang, Liang Lu, and Karen Livescu, "Multitask learning with low-level auxiliary tasks for encoder-decoder based speech recognition," in *Interspeech2017*, 2017, pp. 3532–3536.

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

[18] Kikuo Maekawa, Hanae Koiso, Sadaoki Furui, and Hitoshi Isahara, "Spontaneous speech corpus of Japanese.," in *in International Conference on Language Resources and Evaluation (LREC)*, 2000, pp. 947–9520.

[19] Katunobu Itou, Mikio Yamamoto, Kazuya Takeda, Toshiyuki Takezawa, Tatsuo Matsuoka, Tetsunori Kobayashi, Kiyohiro Shikano, and Shuichi Itahashi, "JNAS: Japanese speech corpus for large vocabulary continuous speech recognition research," *Journal of the Acoustical Society of Japan (E)*, pp. 199–206, 1999.

[20] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, pp. 1929–1958, 2014.

[21] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, pp. 1–15, 2014.

[22] Seiya Tokui, Kenta Oono, Shohei Hido, and Justin Clayton, "Chainer: a next-generation open source framework for deep learning," in *Proceedings of Workshop on Machine Learning Systems (LearningSys) in The Twenty-ninth Annual Conference on Neural Information Processing Systems (NIPS)*, 2015.