AN END-TO-END MODEL FROM SPEECH TO CLEAN TRANSCRIPT FOR PARLIAMENTARY MEETINGS

Masato Mimura*, Shinsuke Sakai*, Tatsuya Kawahara*
* Graduate School of Informatics, Kyoto University, Sakyo-ku, Kyoto, Japan

***Abstract*—This paper presents an end-to-end approach for generating readable and clean text directly from speech signal. While conventional automatic speech recognition (ASR) systems are designed to faithfully reproduce utterances word-by-word, we propose a model that emulates the way a human transcriber/editor creates a clean transcript from speech by skipping fillers, substituting colloquial expressions with more formal ones, inserting punctuation, and performing other types of corrections. An evaluation using 700-hour Japanese Parliamentary speech demonstrates the effectiveness of the proposed approach in generating clean texts suitable for human consumption. We also show that forward-backward decoding and multitask learning leveraging approximate faithful transcripts significantly improve the performance of the direct mapping.**

I. Introduction

Transcribing and archiving meetings, lectures and presentations is one of the important applications for automatic speech recognition (ASR). In order to make a truly useful archive, we need to not only achieve a low recognition error rate, but also consider the readability of system outputs. Since conventional ASR systems are designed to faithfully reproduce all words actually spoken in an utterance, their outputs are not necessarily easy to read and comprehend due to the existence of spoken language phenomena. For example, spontaneous utterances contain not only fillers and disfluencies, but also redundant and colloquial expressions even when fluently spoken. They are often ungrammatical and lack punctuation marks at all. Consequently, a considerable amount of manual edits are required for making final texts appropriate for a written record from faithful transcripts or ASR results [1].

To address this problem, there have been a number of studies on automatic transformation from spoken to written language. They include disfluency detection and removal [2][3], punctuation insertion [4][5][6], and more general speaking style transformation (SST) [7][8][9][10]. A majority of these works rely on machine learning methods such as noisy channel models, CRFs, SVMs, and deep neural networks. These models are typically trained on annotated texts or a parallel corpus of faithful transcripts and corrected texts independently from speech recognition models.

In this paper, we propose an end-to-end (e2e) approach for generating clean texts directly from speech in a manner similar to the way a human transcriber/editor creates a written-style transcript. This direct mapping learns on pairs of speech and the corresponding text from human-made written records, unlike a conventional ASR model which is trained using faithful transcripts as target. To perform this apparently complicated task, we make use of the advantage of an attention-based model that can flexibly attend to only a relevant potion of the input speech to predict the label at each decoding step. Since the proposed model does not require expensive faithful transcripts at all, we can easily build a large amount of training data. Thus, our approach addresses two major problems with the text-based SST: data sparsity and accumulation of errors caused by cascading independently optimized ASR and SST. It is also beneficial to be able to incorporate acoustic information to SST [2][9].

We have been developing a transcription system for the Japanese Parliament (Diet) [11]. In the current version of this system, which has been in official operation from2011, we use a conventional ASR model to generate an initial draft, from which professional editors make final meeting reports. The main aim of this paper is to fundamentally update this transcription system with the proposed e2e approach, in order to reduce the cost and time required in building models and making clean transcripts. A contribution of this paper is that we build and evaluate the model based on a large collection of parliamentary speeches in such a real application scenario, which we believe is informative for developing other transcription systems.

II. Data set

We use a corpus from the House of Representatives of the Diet (national Parliament) of Japan [12]. Parliamentary speeches require a relatively large number of stylistic transformations, and thus are appropriate for evaluating the performance of an SST model [9]. Since the official records of the meetings are open to public[1], we can get the learning target of our model as a ground-truth.

A. Differences between faithful and clean transcripts

Since the written records are made for the purpose of readability and documentation, there is a large difference between what was actually spoken in the meetings and the clean text in the written records, which accounts for 16% of words on average [11]. Fig.1 depicts an example of the pair of a faithful transcript and the corresponding text from the official

[1] http://kokkai.ndl.go.jp/

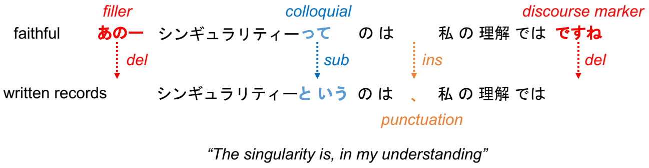
filler
colloquial
discourse marker
faithful
あのー シンギュラリティーって のは 私の理解では ですね
del
sub
ins
del
written records
シンギュラリティーという のは 、 私の理解では
punctuation
"The singularity is, in my understanding"

Fig. 1. An example of the pair of a faithful transcript and the corresponding text in the ofcial written records

records. We can see that a filler word ("e:") and discourse maker (a sentence-end expression "desu ne") were deleted, a colloquial expression was substituted with a formal one (from "tte" to "to iu"), and a comma was inserted. As shown in this example, most of the corrections are represented as simple editing of insertions, deletions or substitutions of one or two words[2]. In general, the major types of edits performed for creating the Parliamentary meeting records are as follows.

a) Deletion: Fillers are completely removed. Discourse markers are often removed, but they can be a part of fluent speech and are kept in some contexts. The earlier part (reparandum) of a repeat or repair is deleted.

b) Substitution: Colloquial expressions are corrected to make a formal sentence.

c) Insertion: Insertion: Punctuation is inserted for improving readability. Function words dropped in spoken language are recovered to make a grammatical sentence.

A more detailed analysis on the same corpus along with the precise occurrence rate of each correction type is found in [9][11].

B. Generation of utterance-level pair data

We need utterance-level pair data of speech and its clean text for training the direct model. For generating the pair data, we first divided the long continuous speech of each meeting into short segments by pauses of longer than 0.2s. This was performed using the short pause segmentation algorithm implemented in the Julius decoder [13]. Then, we identified the corresponding part in the official written records to each speech segment using the following simple procedure.

We decoded each speech segment using a constrained language model (LM), which we will explain in Section 3.2, and a triphone DNN-HMM trained on speech from past meetings. We concatenated the word level recognition results for all segments and inserted a segment boundary token between two consecutive segments. This word sequence was aligned with the full text in the written records of the meeting. The clean text was segmented at word boundaries which corresponded to the segment boundary tokens to extract the target labels for each speech segment.

III. Method

Our goal is to construct a model that directly maps a speech signal to a clean text. For this goal, a model needs to skip regions in the speech that do not have relevant labels (e.g. fillers) and insert tokens which do not have the corresponding acoustic events (e.g. punctuation). Therefore, we adopt an attention-based model among other choices such as a hybrid DNN-HMM [14] or an e2e model based on the CTC loss [15][16]. We also propose two methods to alleviate the difficulty of the direct mapping.

A. Attention-based model

In attention-based speech recognition, we model seq2seq mapping between speech and a label sequence using an encoder-decoder architecture [17][18]. This architecture has two distinct sub-networks. One is the encoder which transforms an acoustic feature sequence to a sequential representation of the same length T. Based on this encoded acoustic information, the other decoder sub-network predicts a label sequence whose length L is usually shorter than the input length T. The decoder uses only a relevant portion of the encoded sequential representation for predicting a label at each time step using the attention mechanism, which is why we adopted the attention-based model for performing the direct generation of a clean text.

More formally, the encoder transforms input acoustic features $X = (x_1, ..., x_T)$ to a sequential representation $H = (h_1, ..., h_T)$ that summarizes the characteristics of the input. In the following decoding step, the hidden state activation of the RNN-based decoder at the l-th time step is computed as:

$$\boldsymbol{r}_l = Recurrency\left(\boldsymbol{r}_{l-1}, \boldsymbol{g}_l, \boldsymbol{y}_{l-1}\right), \tag{1}$$

where g_l and y_{l-1} denote the glimpse at the l-th time step and the predicted label at the previous step, respectively. g_l is a weighted sum of the encoder output sequence as:

$$\begin{aligned} \boldsymbol{g}_l &= \sum_t \alpha_{l,t} \boldsymbol{h}_t, & (2)\\ e_{l,t} &= Score(\boldsymbol{r}_{t-1}, \boldsymbol{h}_t, \boldsymbol{\alpha}_{l-1}), & (3)\\ \alpha_{l,t} &= \exp(e_{l,t}) / \sum_{t'=1}^{T} \exp(e_{l,t'}). & (4) \end{aligned}$$

where $\alpha_{l,t}$ is an attention weight of h_t. Using g_l and r_{l-1}, the decoder predicts the next label y_l as:

$$y_l \sim \boldsymbol{R} \tanh\left(\boldsymbol{P}\boldsymbol{r}_{l-1} + \boldsymbol{Q}\boldsymbol{g}_l\right). \tag{5}$$

In this research, we used words as a recognition unit of the seq2seq model, since it provides the high decoding speed and a simplified architecture [19][20][21].

[2] Human editors also perform more complex corrections which consider the sentence structure or the meaning of the text. However, they are less common and beyond the scope of the proposed scheme.

B. *Direct generation of clean transcript from speech*

We train an attention-based encoder-decoder model using clean texts from the official written records as target. This single model simultaneously performs ASR and all types of corrections described in Section 2.1.

From another point of view, this approach can be considered as an e2e version of the lightly supervised (LSV) training of acoustic models [22][23]. Unlike the conventional LSV, which generates phone labels for training HMM-based models through speech recognition, we make a direct use of written-style texts as the learning target of a seq2seq model. Therefore, our labels are certainly free from recognition errors. Naturally, the proposed approach shares the advantage of the LSV methods that drastically reduces the cost for constructing training data by eliminating the need for expensive faithful transcripts.

We also note that our approach is deeply related to the e2e speech translation that predicts a target language text directly from source language speech [24].

C. *Multitask learning with approximate faithful transcripts*

The direct model needs to perform a more difficult task than either of ASR and text editing. To ease this difficulty, we propose a multi-task learning method to guide the network leveraging the target label sequences for the standard ASR task. Since faithful transcripts are generally not available for a large corpus of spontaneous speech, we exploited the following LSV method [23] to generate an approximation of faithful transcripts.

For every speaker turn in all meetings, we first compute word ngram counts in the corresponding text segment of the written records. These n-grams capture the words and word contexts specific to the particular turn. Then, we convert them to spoken-style so that we can recover spoken language phenomena which are not present in the original clean texts, by applying the LM style transformation based on the framework of statistical machine translation [25] as:

$$P(V) = P(W) \cdot \frac{P(V|W)}{P(W|V)}, \tag{6}$$

where $P(V)$ and $P(W)$ are a spoken-style and written-style n-gram probabilities, respectively. The conditional probabilities $P(V|W)$ and $P(W|V)$ are estimated using a parallel corpus of faithful transcripts and the corresponding clean texts. This translation-based method has a significant advantage that the amount of faithful transcripts required for estimating these probabilities is much smaller than for training LMs from scratch. We actually used a small corpus consisting of only 737K words from meetings held in 2003 in all LSV-related experiments.

Speech recognition using the turn-specific LM obtained from the above procedure can recover the faithful transcript of an utterance with a high accuracy. This recognition result is provided to the additional output layer for the ASR subtask in order to help the convergence of the direct mapping task. We specifically used character-level transcripts and the CTC loss for this subtask to promote the efficiency of the MTL [21][26].

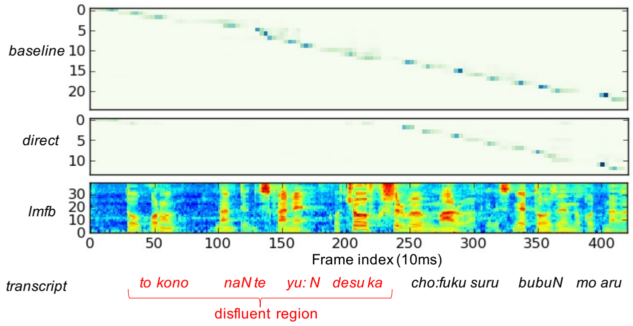

Fig. 2. Attention weights from the baseline and the proposed direct model for an utterance with a long disfluent region

D. *Forward-backward decoding*

The standard attention-based ASR model decodes from the start toward the end of an utterance mostly based on acoustic information in a unidirectional way. In contrast, we should perform text editing or rewriting considering the whole structure of a complete sentence. For example, the comma in Fig.1 was inserted to separate the subject and its modier part of the sentence. Clearly, this comma insertion could not be performed without recognizing the following several words. This implies that right contexts are as important as left contexts in text correction.

By taking this consideration into account, we exploit the forward-backward attention decoder [27] to improve the correction performance. In this method, speech is decoded not only from left-to-right, but also from right-to-left using a dedicated backward decoder. Partial sentence candidates from both decoders are concatenated based on the estimated occurrence time of each word to generate a new complete sentence. Among all candidates, the best hypothesis is searched for according to the combined probabilities of forward and backward decoding. A more detailed description of the algorithm is in [27]. We originally proposed this bidirectional decoding for improving the attention-based ASR. However, it is potentially more effective for the direct mapping to a clean text, since it is crucial to look at both of left and right contexts in editing text.

IV. EXPERIMENTAL EVALUATION

We evaluated the proposed approach on the Parliamentary speeches from the House of Representatives of the Diet of Japan recorded in 2015. The data were divided into the training set consisting of 708-hour speech from 14 plenary sessions and 194 committee meetings, and the test set consisting of 20-hour speech from 5 committee meetings with a wide variety of topics. We built two baseline ASR models: an e2e model based on the CTC loss and an attention-based seq2seq model. These ASR models were trained on the labels generated by

TABLE I
CHARACTER ERROR RATES BETWEEN SYSTEM OUTPUTS AND THE OFFICIAL WRITTEN RECORDS (%)

	error type			
model	del	sub	ins	total
CTC ASR	5.1	5.2	12.0	22.2
attn. ASR	3.8	5.7	14.6	22.5
+ filler word removal	4.5	4.8	7.4	16.7
attn. ASR + attn. SST (cascade)	5.3	4.7	5.2	15.2
CTC direct	6.8	4.3	2.7	13.8
attn.direct	**4.0**	**3.7**	**3.1**	**10.8**
+ MTL	**3.9**	**3.6**	**2.8**	**10.3**
+ forward-backward decoding	**3.2**	**3.3**	**2.7**	**9.2**

the LSV method in Section 3.3. We built the proposed direct model using an encoder-decoder architecture as described in Section 3. For comparison, we also trained a direct mapping model using the CTC loss. We used Pytorch [28] to train all models.

We implemented the acoustic encoder in all attention models with a 5-layer bidirectional LSTM [29], while the decoder consists of a one-layer unidirectional LSTM and a softmax output layer. Similarly, the CTC models consist of a 5-layer bidirectional LSTM and a softmax output layer. In the MTL method, we added an CTC output layer for the ASR auxiliary task on the top of the shared encoder in the attention-based direct model. All LSTM layers have 320 memory cells. We used label smoothing [30] to improve the optimization in training the attention models. The vocabulary sizes of the ASR baseline and the direct model are 21,455 and 21,573, respectively. A 40-dimensional vector consisting of 40-channel log Mel-scale filter-bank (lmfb) outputs was used as the acoustic feature.

We used character error rates between the system outputs and the text of the official written records as a metric for the performance of generating a clean text from speech. The results of all models are shown in TABLE I. Note that we purposely excluded punctuation marks in calculation of these error rates, because the ASR baselines have no chance to insert punctuation.

A. Baseline ASR vs. direct model

Both ASR baselines gave high insertion error rates, which reflects the fact that about half of all corrections performed in making the written records are categorized as removal of fillers [11]. Simply removing lexical fillers halved the insertion errors of the ASR model (14.6% to 7.4%).

In contrast, the attention-based direct model yielded a very low insertion error rate without any postprocessing (3.1%). To illustrate this characteristic, Fig.2 depicts the attention weights from the baseline ASR and the direct model for an utterance in the test set with a long disfluent region. We can see that the direct model successfully ignored all disfluencies, while the baseline attended to all regions in the utterance. The direct model significantly reduced not only insertion errors but also substitution errors, and yielded a much shorter edit distance to the reference clean text than the baseline ASR. It is also interesting to see that the CTC model performed less well than the attention model in the direct mapping task, while they gave similar performances when used as standard ASR models. We see that the attention model works more flexibly with omitted and additional tokens than the CTC.

B. Cascaded ASR and text-based SST vs. direct model

We also implemented and evaluated an attention-based modular method as a reasonable alternative of the conventional approach, which cascades ASR and text-based SST. In this method, we performed ASR and SST using two separate seq2seq models. The text-based SST was trained using the LSV transcripts as input and the clean text as target. In the runtime, this SST takes the output of the ASR model as input.

We found two interesting phenomena with the result of this cascade approach. On one hand, it gave a much fewer insertion errors than the baseline ASR followed by filler removal (7.4% vs. 5.2%). This suggests that the text-based SST implemented with a seq2seq model can perform more various types of corrections than simply removing filler words. On the other hand, it was significantly worse than the proposed direct model in terms of all types of error rates. This confirms that the direct approach is much more effective than cascading isolated ASR and SST modules.

The MTL method further improved the performance of the direct model by 0.5 points. This shows that we can mitigate the difficulty of the direct mapping task by incorporating an easier auxiliary task. Furthermore, the forward-backward decoding gave an additional large improvement of 1.1 points, which clearly demonstrates the importance of right contexts in correcting texts. These improvements with the MTL and bidirectional decoding are statistically significant at the 1% level. The best direct model yielded even a higher character correctness (complement of the sum of deletion and insertion errors) than the state-of-the-art hybrid DNN-HMM equipped with a large LM and lexicon with 67K word entries (93.5% vs. 92.7%) with a decoding speed faster by a factor of 50.

It is important to emphasize that the aim of this section is not to make a fair comparison between the direct and cascade approaches[3], but to demonstrate that accurate generation of clean texts is possible for real parliamentary meetings based on our method without requiring faithful transcripts at all.

C. Punctuation insertion

We cannot tell apart ASR errors and correction errors from the results in TABLE I. Here, we compare the pure correction performance of the cascade and direct models by showing their F-measure for punctuation insertion in TABLE II. While the cascade model gave a reasonable F-measure for comma and period insertion, the direct model gave much better results. By

[3] Ideally, the SST model should be trained using the faithful transcripts for all training data as input, but it is unrealistically expensive. The attention-based model cannot be reliably trained with the small corpus mentioned in Section 3.3.

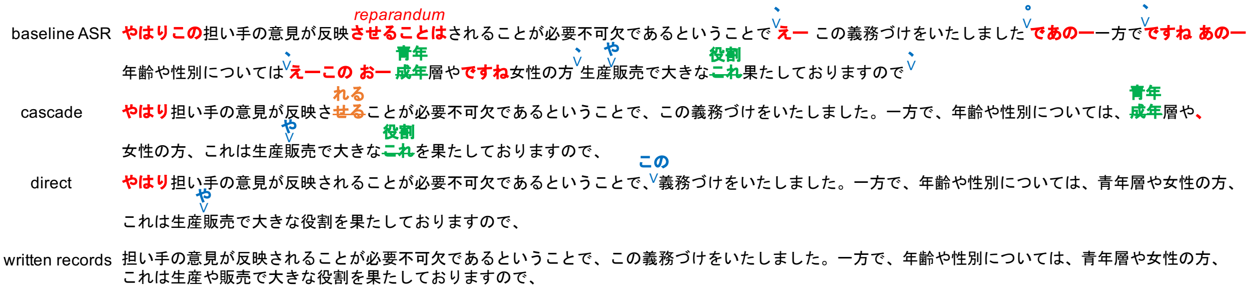
reparandum
baseline ASR
やはりこの担い手の意見が反映させることはされることが必要不可欠であるということで、えー この義務づけをいたしました。であのー一方で、ですね あのー
、 青年 、 や 役割 、
年齢や性別については、えーこの おー ~~成年~~層やですね女性の方、や生産販売で大きな~~これ~~果たしておりますので、
cascade
れる 青年
やはり担い手の意見が反映さ~~せる~~ことが必要不可欠であるということで、この義務づけをいたしました。一方で、年齢や性別については、~~成年~~層や、
や 役割
女性の方、これは生産販売で大きな~~これ~~を果たしておりますので、
direct
この
やはり担い手の意見が反映されることが必要不可欠であるということで、義務づけをいたしました。一方で、年齢や性別については、青年層や女性の方、
や
これは生産販売で大きな役割を果たしておりますので、
written records
担い手の意見が反映されることが必要不可欠であるということで、この義務づけをいたしました。一方で、年齢や性別については、青年層や女性の方、
これは生産や販売で大きな役割を果たしておりますので、

Fig. 3. An example of the system output and the corresponding clean text. Characters which should be deleted, inserted and substituted in the written records appear in the red, blue and orange fonts, respectively. Those in the green font are recognition errors.

TABLE II
F-MEASURE FOR PUNCTUATION INSERTION (%)

model	comma	period
attn. ASR + attn. SST (cascade)	0.711	0.750
attn. direct	**0.732**	**0.827**
+ MTL	**0.740**	**0.829**
+ forward-backward decoding	**0.753**	**0.834**

using the MTL and bidirectional decoding, the performance is further improved, following a parallel trend to the results in Table I. Since a quantitative analysis on other types of corrections requires additional annotations, it is left for future work.

Fig.3 compares the output of the baseline ASR, the cascade model, and the direct model enhanced by the MTL and bidirectional decoding, along with the corresponding ground-truth text from the official written records for an example utterance. We can see that the direct model significantly reduced the number of edits required to modify the system output into the reference. We also note that in the output of the cascade approach, two recognition errors made by the baseline ASR remain uncorrected, while the direct model is free from these recognition errors. It is also notable that the direct model successfully removed only the reparandum part of the repair. Overall, we observe the proposed direct model does not only perform appropriate corrections, but also reduce recognition errors. This may be due to two reasons: the clean text target does not include recognition errors unlike the conventional LSV, and it is much more linguistically constrained and has lower perplexity than faithful transcripts.

V. CONCLUSION

We here proposed an e2e approach for directly mapping speech to a clean text and evaluated on a large corpus of Parliamentary speech. We showed the attention-based model is flexible enough to perform this complicated task with a low error rate. We also demonstrated the effectiveness of an MTL method leveraging the standard ASR subtask and forward-backward decoding. From these encouraging results, we confirmed that our method can drastically reduce the cost and time in making written records of Parliamentary meetings. The clean text output is also useful for downstream processing such as machine translation [31].

REFERENCES

[1] D.Jones, F.Wolf, E.Gibson, E.Williams, E.Fedorenko, D.Reynolds, and M.Zissman, “Measuring the readability of automatic speech-to-text transcripts,” in *Eurospeech*, 2003, pp. 1585–1588.

[2] Y. Liu, E. Shriberg, A. Stolcke, D. Hillard, M. Ostendorf, , and M. Harper, “Enriching speech recognition with automatic de- tection of sentence boundaries and disfluencies,” *IEEE Trans. Audio, Speech & Language Process.*, vol. 14, pp. 1526–1540, 2006.

[3] J. Yeh and C. Wu, “Edit disfluency detection and correc- tion using a cleanup language model and an alignment model,” *IEEE Trans. Audio, Speech & Language Process.*, vol. 14, pp. 1574–1583, 2006.

[4] M. Paulik, S. Rao, I. Lane, S. Vogel, and T. Schultz, “Sentence segmentation and punctuation recovery for spoken language translation,” in *Proc. ICASSP*, 2008.

[5] A. Gravano, M. Jansche, and M. Bacchiani, “Restoring punctuation and capitalization in transcribed speech,” in *Proc. ICASSP*, 2009, pp. 4741–4744.

[6] Y. Akita and T. Kawahara, “Automatic comma insertion of lecture transcripts based on multiple annotations,” in *INTERSPEECH*, 2011, pp. 2889–2892.

[7] T. Hori, D. Willett, and Y. Minami, “Paraphrasing spontaneous speech using weighted finite-state transducers,” in *ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition*, 2013.

[8] K. Shitaoka, H. Nanjo, and T. Kawahara, “Automatic transformation of lecture transcription into document style using statistical framework,” in *INTERSPEECH*, 2004, pp. 2169–2172.

[9] G.Neubig, Y.Akita, S.Mori, and T.Kawahara, “A monotonic statistical machine translation approach to speaking style transformation,” in *Computer Speech and Language*, vol. 26, 2012, pp. 349–370.

[10] R. Sproat and N. Jaitly, “An rnn model of text normalization,” in *INTERSPEECH*, 2017, pp. 754–757.

[11] T.Kawahara, “Automatic meeting transcription system for the Japanese Parliament (Diet),” in *APSIPA*, 2017.

[12] M. Y.Akita and T.Kawahara, “Automatic transcription system for meetings of the japanese national congress,” in *INTERSPEECH*, 2009, pp. 84–87.

[13] A. Lee, T. Kawahara, and K. Shikano, “Julius : an open source real-time large vocabulary recognition engine,” in *EUROSPEECH*, pp. 1691–1694,.

[14] G.E.Hinton, L.Deng, D.Yu, G.Dahl, A.Mohamed, N.Jaitly, A.Senior, V.Vanhoucke, P.Nguyen, T.Sainath, and B.Kingsbury, “Deep neural networks for acoustic modeling in speech recognition,” *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[15] A. Graves, S. Fernandez, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification : Labelling unsegmented sequence data with recurrent neural networks,” in *Proc. of the 23st International Conference on Machine Learning*, 2006, pp. 369–376.

[16] H. Sak, A. Senior, K. Rao, and F. Beaufays, “Fast and accurate recurrent neural network acoustic models for speech recognition,” in *arXiv preprint arXiv:1607.06947*, 2015.

[17] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, “Attention-based models for speech recognition,” in *Advances in Neural Information Processing Systems (NIPS)*, 2015, pp. 577–585.

[18] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, “Listen, attend and spell: A neural network for large vocabulary conversational speech recognition,” in *ICASSP*, 2016, pp. 4960–4964.

[19] K. Audhkhasi, B. Ramabhadran, G. Saon, M. Picheny, and D. Nahamoo, “Direct acoustics-to-word models for English conversational speech recognition,” in *Interspeech*, 2017, pp. 959–963.

[20] H. Soltau, H. Liao, and H. Sak, “Neural speech recognizer: Acoustic-to-word LSTM model for large vocabulary speech recognition,” in *Interspeech*, 2017, pp. 3707–3711.

[21] S. Ueno, H. Inaguma, M. Mimura, and T. Kawahara, “Acoustic-to-word attention-based model complemented with character-level CTC-based model,” in *ICASSP*, 2018.

[22] L.Lamel, J.Gauvain, and G.Adda, “Investigating lightly supervised acoustic model training,” in *ICASSP*, vol. 1, 2001, pp. 477–480.

[23] T.Kawahara, M.Mimura, and Y.Akita, “Language model transformation applied to lightly supervised training of acoustic model for congress meetings, booktitle =.”

[24] Ron.J.Weiss, J. Chorowski, N. Jaitly, Y. Wu, and Z. Chen, “Sequence-to-sequence models can directly translate foreign speech,” in *INTERSPEECH*, 2017, pp. 2625–2629.

[25] Y.Akita and T.Kawahara, “Topic-independent speaking-style transformation of language model for spontaneous speech recognition,” in *ICASSP*, vol. 4, 2007, pp. 33–36.

[26] T. H. Suyoun Kim and S. Watanabe, “Joint ctc- attention based end-to-end speech recognition using multi-task learning,” in *ICASSP*, 2017, pp. 4835–4839.

[27] M. Mimura, S. Sakai, and T. Kawahara, “Forward-backward attention decoder,” in *INTERSPEECH*, 2018, pp. 2232–2236.

[28] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, “Automatic differentiation in pytorch,” 2017.

[29] S.Hochreiter and J.Schmidhuber, “Long Short-Term Memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[30] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[31] S. Rao, I. Lane, and T. Schultz, “Improving spoken language translation by automatic disfluency removal: evidence from conversational speech transcripts,” in *Machine Translation Summit XI*, 2007, pp. 177–180.