Disfluency Detection Based on Speech-Aware Token-by-Token Sequence Labeling with BLSTM-CRFs and Attention Mechanisms

Tomohiro Tanaka, Ryo Masumura, Takafumi Moriya, Takanobu Oba, Yushi Aono
NTT Media Intelligence Laboratories, NTT Corporation, Japan
E-mail: tomohiro.tanaka.ht@hco.ntt.co.jp

***Abstract*—This paper presents a new method for token-by-token sequence labeling that can leverage not only lexical information but also speech information without any alignments. Our motivation is to detect disfluencies such as fillers and word fragments robustly from spontaneous speech. Disfluency detection is often modeled as a token-by-token sequence labeling using a transcribed text via automatic speech recognition. However, utilizing the lexical information alone is not sufficient because the disfluencies cause changes to speech information. One problem is that the speech and the transcribed text need to be aligned when we handle speech and lexical information simultaneously. This prevents introducing speech information to the disfluency detection. To solve this problem, we propose a method for token-by-token sequence labeling, one that can simultaneously use lexical and speech information without requiring any alignments. To this end, we introduce attention mechanisms into a method for neural sequence labeling based on bi-directional long short-term memory recurrent neural network conditional random fields. The attention mechanisms enable us to find the term of disfluencies from speech automatically. Our experimental results show that the proposed method using acoustic and prosodic features improves the labeling accuracy compared with that using lexical features alone.**

Index Terms: disfluency detection, sequence labeling, BLSTM-CRFs

I. Introduction

Automatic speech recognition (ASR) systems have been making great progress and their performance indicates they have high practical value. ASR systems usually transcribe speech only for utterance content, but spontaneous speech as a whole includes rich information. For example, spontaneous speech often includes disfluencies such as fillers and word fragments. By giving the labels of these disfluencies to the texts of ASR output, we can delete unnecessary parts in the texts and leverage the labels for post-applications including spoken dialog, speech translation, and speech summarization.

The purpose of this study is to detect disfluencies from spontaneous speech robustly and to label them to transcribed text from ASR, as shown in Figure 1. Disfluency detection is often formulated as a token-by-token sequence labeling using the text. Conventional methods of disfluency detection are to use conditional random fields (CRFs)-based classifiers [1]–[4]. A recurrent neural network-based approaches [5], [6] have been proposed, and they have provided better performance than the conventional methods. On the other hand, bi-directional long short-term memory recurrent neural network-CRFs (BLSTM-CRFs) have provided better performance than BLSTM in several tasks such as named entity recognition [7] and part-of-speech tagging [8]. BLSTM enables the model to capture past and future contexts in the text. CRFs is utilized for jointly decoding on top of BLSTM to consider the relationship between output labels. However, with disfluency detection, the lexical information alone is not sufficient because the disfluencies cause changes to speech information. For example, it is assumed that ambiguous phonemes appear in the utterance including word fragments and fillers cause changes to prosodic information.

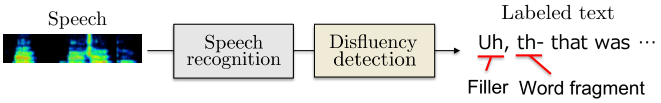

Fig. 1. Labeling disfluencies with automatic speech recognition.

Unfortunately, the speech and the transcribed text need to be aligned for handling them simulataneously. Previous studies utilized segmented prosodic information for disfluency detection [1], [3]. In other tasks of token-by-token sequence labeling with speech information, Klejch et al. [9] proposed an approach for punctuation estimation, one in which speech and lexical features are simultaneously used with a neural hierarchical encoder network where the token-level alignments are given. Wang et al. [10] used a different neural network topology for discourse marker detection after deciding the alignments. These studies needed to obtain the alignments with speech and text. This made it difficult to introduce speech information to the disfluency detection.

In this paper, we propose a method for token-by-token sequence labeling that can simultaneously utilize lexical and speech information without requiring any alignments between speech and text. In order to use the speech information, our idea is to introduce attention mechanisms [11]–[13] into a method for neural sequence labeling based on BLSTM-CRFs. The attention mechanisms automatically find the terms of disfluencies from speech. Our proposed method leverage lexical

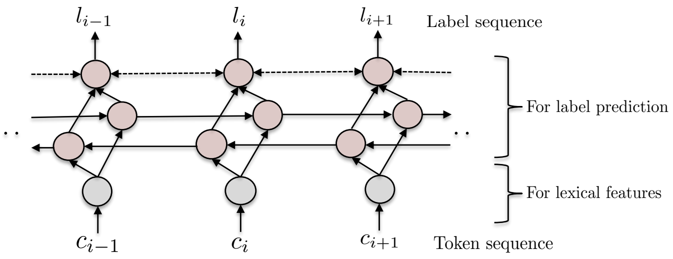

Fig. 2. Token-by-token sequence labeling with BLSTM-CRFs. This figure is an example of labeling to a token c_i.

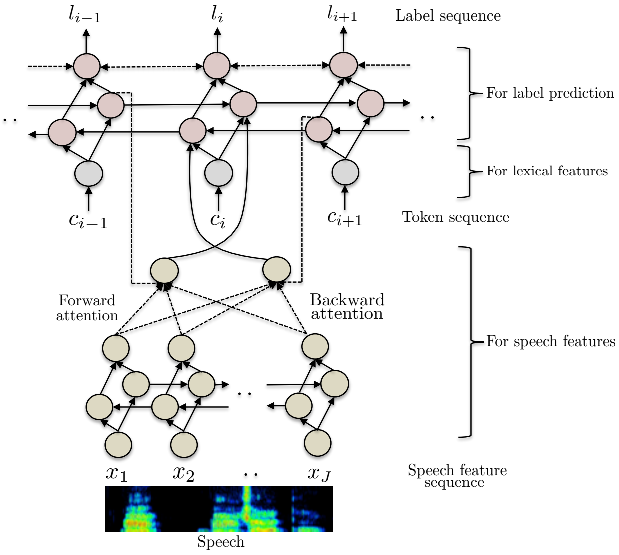

Fig. 3. Speech-aware text-based sequence labeling with BLSTM-CRFs and attention mechanisms. This figure is an example of labeling to a token c_i.

and speech information efficiently for detecting disfluencies and labeling them to texts.

We evaluate our proposed method with using a Japanese lecture task with the Corpus of Spontaneous Japanese (CSJ) [14]. Experimental evaluations show that the attention mechanisms enable the effective use of acoustic and prosodic features. We found that the speech features increase the accuracy of the disfluency detection.

This paper is organized as follows. Section 2 describes a method of token-by-token sequence labeling based on BLSTM-CRFs. Section 3 details our proposed method based on BLSTM-CRFs and attention mechanisms. The experiments are shown in Section 4. Section 5 concludes the paper.

II. Token-by-Token Sequence Labeling with BLSTM-CRFs

In this section, we describe token-by-token sequence labeling with BLSTM-CRFs. The task of token-by-token sequence labeling is finding the most probable label sequence $\hat{\boldsymbol{l}} = \{l_1, l_2 \cdots, l_i, \cdots, l_I\}$ given a token sequence $\boldsymbol{c} = \{c_1, c_2, \cdots, c_i, \cdots, c_I\}$. This problem is formulated as

$$\hat{\boldsymbol{l}} = \arg\max_{\boldsymbol{l}} P(\boldsymbol{l}|\boldsymbol{c};\boldsymbol{\Theta}), \tag{1}$$

where $\boldsymbol{l} = \{l_1, l_2 \cdots, l_i, \cdots, l_I\}$ denotes a label sequence, and where $\boldsymbol{\Theta}$ is the parameters of a model for token-by-token sequence labeling.

BLSTM-CRFs is known as a model for token-by-token sequence labeling. Figure 2 illustrates BLSTM-CRFs, which estimate a label l_i for a token c_i. In BLSTM, each token c_i in a token sequence $\boldsymbol{c}$ is encoded to 1-of-K representation and embedded into a continuous representation as

$$\boldsymbol{d}_i = \texttt{EMBED}(c_i, \boldsymbol{\theta}_d), \tag{2}$$

where $\texttt{EMBED}(\cdot)$ is a function that converts a token into a distributed representation, and where $\boldsymbol{\theta}_d$ is a trainable parameter. Then, embedded vectors are input to bi-directional LSTM as

$$\overrightarrow{\boldsymbol{h}}_i = \overrightarrow{\texttt{LSTM}}(\boldsymbol{d}_i, \overrightarrow{\boldsymbol{h}}_{i-1}, \boldsymbol{\theta}_{lf}), \tag{3}$$

$$\overleftarrow{\boldsymbol{h}}_i = \overleftarrow{\texttt{LSTM}}(\boldsymbol{d}_i, \overleftarrow{\boldsymbol{h}}_{i+1}, \boldsymbol{\theta}_{lb}), \tag{4}$$

where $\overrightarrow{\texttt{LSTM}}(\cdot)$ and $\overleftarrow{\texttt{LSTM}}(\cdot)$ represent LSTM functions of forward and backward LSTM. $\boldsymbol{\theta}_{lf}$ and $\boldsymbol{\theta}_{lb}$ are the trainable model parameters. The hidden state $\boldsymbol{h}_j$ is calculated by concatenating $\overrightarrow{\boldsymbol{h}}_j$ and $\overleftarrow{\boldsymbol{h}}_j$ as

$$\boldsymbol{h}_i = [\overrightarrow{\boldsymbol{h}}_i^\top, \overleftarrow{\boldsymbol{h}}_i^\top]^\top. \tag{5}$$

The concatenated vector $\boldsymbol{h}_i$ is converted into a vector as

$$\boldsymbol{o}_i = g(\boldsymbol{h}_i, \boldsymbol{\theta}_o), \tag{6}$$

where $g(\cdot)$ is the function of linear transformation and $\boldsymbol{\theta}_o$ is the model parameter. The conditional probability of $\boldsymbol{l}$ is calculated as

$$P(\boldsymbol{l}|\boldsymbol{c};\boldsymbol{\Theta}) = \frac{\prod_{i=1}^{I} \varphi(l_{i-1}, l_i, \boldsymbol{o}_i; \boldsymbol{\theta}_p)}{\sum_{\bar{\boldsymbol{l}}} \prod_{i=1}^{n} \varphi(\bar{l}_{i-1}, \bar{l}_i, \boldsymbol{o}_i; \boldsymbol{\theta}_p)}, \tag{7}$$

where $\varphi(\cdot)$ represents the function to multiply weight parameters and outputs of LSTM layers with an exponential function, and where $\boldsymbol{\theta}_p$ is the parameter. Finally, a label sequence is obtained to maximize the probability as Eq. (7). The most probable label sequence can be obtained through the Viterbi algorithm.

The trainable parameters can be summarized as

$$\boldsymbol{\Theta} = \{\boldsymbol{\theta}_d, \boldsymbol{\theta}_{lf}, \boldsymbol{\theta}_{lb}, \boldsymbol{\theta}_o, \boldsymbol{\theta}_p\}. \tag{8}$$

The parameters are optimized to maximize the probability of correct label sequence $\boldsymbol{l}$ giving token sequence $\boldsymbol{c}$.

$$\mathcal{L}(\boldsymbol{\Theta}) = -\sum_{(\boldsymbol{l}', \boldsymbol{c}') \in \mathcal{D}} \log P(\boldsymbol{l}'|\boldsymbol{c}';\boldsymbol{\Theta}), \tag{9}$$

where $\mathcal{D}$ is training data. The training data $\mathcal{D}$ can be described as

$$\mathcal{D} = \{(\boldsymbol{c}_1, \boldsymbol{l}_1), (\boldsymbol{c}_2, \boldsymbol{l}_2), \ \cdots, (\boldsymbol{c}_N, \boldsymbol{l}_N)\}, \tag{10}$$

where N is the number of pairs of tokens and labels in the training data.

III. Speech-Aware Token-by-Token Sequence Labeling

The task given in this study is finding the most probable label sequence $\hat{\boldsymbol{l}}$ given an speech feature sequence $\boldsymbol{x} = \{x_1, x_2 \cdots, x_j, \cdots, x_J\}$ and a token sequence $\boldsymbol{c}$. This problem is formulated as

$$\hat{\boldsymbol{l}} = \arg\max_{\boldsymbol{l}} P(\boldsymbol{l}|\boldsymbol{c}, \boldsymbol{x}; \boldsymbol{\Theta}), \tag{11}$$

where $\boldsymbol{l} = \{l_1, l_2 \cdots, l_i, \cdots, l_I\}$ denotes a label sequence, and where $\boldsymbol{\Theta}$ is the parameters of a model for speech-aware token-by-token sequence labeling.

Figure 3 illustrates a model of the speech-aware token-by-token sequence labeling, which estimates a label l_i for a token c_i. The model has three networks, which are a network for speech features, lexical features, and sequence labeling. In the network for speech features, the speech feature sequence $\boldsymbol{x} = \{x_1, x_2 \cdots, x_j, \cdots, x_J\}$ is input to the BLSTM as

$$\overrightarrow{\boldsymbol{s}}_j = \overrightarrow{\texttt{LSTM}}(\boldsymbol{x}_j, \overrightarrow{\boldsymbol{s}}_{j-1}, \boldsymbol{\theta}_{lf}), \tag{12}$$

$$\overleftarrow{\boldsymbol{s}}_j = \overleftarrow{\texttt{LSTM}}(\boldsymbol{x}_j, \overleftarrow{\boldsymbol{s}}_{j+1}, \boldsymbol{\theta}_{lb}), \tag{13}$$

where $\boldsymbol{\theta}_{lf}$ and $\boldsymbol{\theta}_{lb}$ are the trainable model parameters. The hidden state $\boldsymbol{s}_j$ is calculated by concatenating $\overrightarrow{\boldsymbol{s}}_j$ and $\overleftarrow{\boldsymbol{s}}_j$ as

$$\boldsymbol{s}_j = [\overrightarrow{\boldsymbol{s}}_j^\top, \overleftarrow{\boldsymbol{s}}_j^\top]^\top. \tag{14}$$

In the network for lexical features, Distributed representation $\boldsymbol{d}_i$ is calculated in the network for lexical features using the weight matrix given target sentence $\boldsymbol{c} = \{c_1, c_2, \cdots, c_i, \cdots, c_I\}$ as

$$\boldsymbol{d}_i = \texttt{EMBED}(c_i, \boldsymbol{\theta}_d). \tag{15}$$

The hidden state is calculated in the network for labeling using the LSTM function as

$$\overrightarrow{\boldsymbol{h}_i} = \overrightarrow{\texttt{LSTM}}([\boldsymbol{d}_i, \overrightarrow{\boldsymbol{v}}_i], \overrightarrow{\boldsymbol{h}}_{i-1}, \boldsymbol{\theta}_{fs}), \tag{16}$$

$$\overleftarrow{\boldsymbol{h}_i} = \overleftarrow{\texttt{LSTM}}([\boldsymbol{d}_i, \overleftarrow{\boldsymbol{v}}_i], \overleftarrow{\boldsymbol{h}}_{i+1}, \boldsymbol{\theta}_{bs}), \tag{17}$$

where $\overrightarrow{\boldsymbol{v}}_i]$ and $\overleftarrow{\boldsymbol{v}}_i]$ are the context vector constructed in each input token. The context vector $\overrightarrow{\boldsymbol{v}_i}$ for the forward direction is caclulated as

$$\overrightarrow{\boldsymbol{v}}_i = \sum_{j=1}^{J} \overrightarrow{\alpha}_{j,i} \overrightarrow{\boldsymbol{s}}_j, \tag{18}$$

where the attention weight $\overrightarrow{\alpha}_{j,i}$ is calculated as

$$\overrightarrow{\alpha}_{j,i} = \frac{\exp(\overrightarrow{e}_{j,i})}{\sum_{j=1}^{J} \exp(\overrightarrow{e}_{j,i})}. \tag{19}$$

In this study, we investigated two types of attention mechanisms: content-based [11] and location-based [13] attention. In addition, we used uniform distribution for the attention weight as a comparison ("No-attention"). For each method, $\overrightarrow{e}_{j,i}$ in Eq. (19) is calculated as

$$\begin{cases} \text{Location-based attention :} \\ \overrightarrow{\boldsymbol{f}}_j = \overrightarrow{\boldsymbol{F}} * \overrightarrow{\boldsymbol{\alpha}}_{j-1}, \\ \overrightarrow{e}_{j,i} = \texttt{Score}(\overrightarrow{\boldsymbol{s}}_i, \overrightarrow{\boldsymbol{h}}_j, \overrightarrow{\boldsymbol{f}}_{j,i}, \boldsymbol{\theta}_{fe}), \\ \\ \text{Content-based attention :} \\ \overrightarrow{e}_{j,i} = \texttt{Score}(\overrightarrow{\boldsymbol{s}}_i, \overrightarrow{\boldsymbol{h}}_j, \boldsymbol{\theta}_{fe}), \\ \\ \text{No-attention :} \\ \overrightarrow{e}_{j,i} = 0, \end{cases} \tag{20}$$

where $\texttt{Score}(\cdot)$ is the nonlinear function with additive operations, where $\overrightarrow{\boldsymbol{s}}_i$ is the hidden state of the forward direction in the labeling network, "$*$" indicates the convolutional function, and where $\boldsymbol{F}$ and $\boldsymbol{\theta}_{fe}$ are the trainable model parameters. For the backward direction, $\overleftarrow{\boldsymbol{v}_i}$, $\overleftarrow{\alpha}_{j,i}$, and $\overleftarrow{\boldsymbol{e}}_{j,i}$ are calculated in the same manner as Eq. (18-20). The hidden state $\boldsymbol{h_i}$ is calculated by concatenating $\overrightarrow{\boldsymbol{h}}_i$ and $\overleftarrow{\boldsymbol{h}}_i$ as

$$\boldsymbol{h}_i = [\overrightarrow{\boldsymbol{h}}_i^\top, \overleftarrow{\boldsymbol{h}}_i^\top]^\top. \tag{21}$$

The concatenated vector $\boldsymbol{h}_i$ is converted into a vector as

$$\boldsymbol{o_i} = g(\boldsymbol{h}_i, \boldsymbol{\theta}_o), \tag{22}$$

where $\boldsymbol{\theta}_o$ is the model parameter. Given the symbol sequence $\boldsymbol{c} = \{c_1, \cdots, c_I\}$, the probability of $\boldsymbol{l}$ is calculated as

$$P(\boldsymbol{l}|\boldsymbol{c}, \boldsymbol{x}; \boldsymbol{\Theta}) = \frac{\prod_{i=1}^{I} \varphi(l_{i-1}, l_i, \boldsymbol{o}_i; \boldsymbol{\theta}_p)}{\sum_{\bar{\boldsymbol{l}}} \prod_{i=1}^{n} \varphi(\bar{l}_{i-1}, \bar{l}_i, \boldsymbol{o}_i; \boldsymbol{\theta}_p)}, \tag{23}$$

where $\boldsymbol{\theta}_p$ is the parameter. Finally, the most probable label sequence is obtained to maximize the probability as Eq. (23.) The label sequence can be obtained through the Viterbi algorithm.

The trainable parameters are summarized as

$$\boldsymbol{\Theta} = \{\boldsymbol{\theta}_{lf}, \boldsymbol{\theta}_{lb}, \boldsymbol{\theta}_{fe}, \boldsymbol{\theta}_{be}, \boldsymbol{\theta}_{d}, \boldsymbol{\theta}_{fs}, \boldsymbol{\theta}_{bs}, \boldsymbol{\theta}_{o}, \boldsymbol{\theta}_{p}, \boldsymbol{F}\}. \tag{24}$$

In the training, they are updated to maximize the conditional probability of the correct label when giving speech features and tokens. Thus, the model parameters are optimized by maximizing the probabilities as

$$\mathcal{L}(\boldsymbol{\Theta}) = -\sum_{(\boldsymbol{l}', \boldsymbol{c}', \boldsymbol{x}') \in \mathcal{D}} \log P(\boldsymbol{l}'|\boldsymbol{c}', \boldsymbol{x}'; \boldsymbol{\Theta}), \tag{25}$$

where $\mathcal{D}$ is the sets of labels, tokens, and speech features. Unlike the method in Section II, speech features are given in this problem. We used acoustic information and prosodic information as the speech features $\boldsymbol{x}'$. We assumed that acoustic and prsodic information enable us to capture ambiguous phonemes appear in the utterance including word fragments and fillers cause changes to prosodic information.

IV. Experiments

A. Setups

1) Data: We used Japanese lecture corpus of the CSJ to evaluate our models with disfluency sequence labeling of begin, inside, and outside labels. speech and its manual transcription with disfluency labels. Table I shows examples of the

TABLE I

Example of BIO labels of a sentence in Japanese dataset. “今日の天気は晴れ” means “It’s sunny today”. The target labels were begin-filler (B-F), inside-filler (I-F), begin-word fragment (B-WF), indide-word fragment (I-WF) and other (O). “えー” is one of the Japanese fillers and “きょ” is a word fragment derived from the word “今日”.

Text	え	ー	き	ょ	今	日	の	天	気	は	晴	れ
BIO labels	B-F	I-F	B-WF	I-WF	O	O	O	O	O	O	O	O

TABLE II

Details of training, development, and test data

Data	# of characters	# of sentences	Hours
Training	5M	149K	268
Development	53K	2K	3
Test	88K	3K	5

TABLE III

The number of disfluencies in training, development and test data.

Data	Fillers	Word fragments
Training	203K	42K
Development	2029	536
Test	2749	701

BIO labels in a sentence of the CSJ. We used character-level labeled texts because they are independent of morphological analysis and have a small vocabulary. The test data were CSJ standard evaluation set 1 and 2. The target disfluency labels were fillers and word fragments. Table II shows the details of the training, development and test data and Table III shows the number of BIO labels in the dataset. Because the number of word fragments was much lower than the number of fillers, the detection of word fragments was a more difficult task.

2) Models: All sequence labeling models predicted the labels for each character in a sentence. The vocabulary size of our models was 2748 characters, which corresponded to the input dimension of the network for lexical information. The BLSTM in the network of label prediction had three hidden layers and 320 LSTM units in each layers and each directions. When we used speech features, the network for speech features had four hidden layers and 320 LSTM units in each layers and each directions. The content-based or location-based attention was used for forward and backward directions. The number of the target labels was five which are begin-filler (B-F), inside-filler (I-F), begin-word fragment (B-WF), indide-word fragment (I-WF) and other (O). We used the AdaDelta algorithm [15] to optimize the model parameters and performed early stopping using the labeling accuracy of the development set.

3) Input features: We prepared lexical, acoustic, prosodic features as input speech information for speech-aware token-by-token sequence labeling. We used $1-of-K$ representation of characters for the lexical features. We used 40 mel-scale filter-bank features and their delta and delta-delta as acoustic features. The acoustic features are totaly 120-dimensional vectors in each time frame. We used 3-dimensional prosodic features that are a warped normalized cross correlation function, log-pitch with a probability of voicing-weighted mean subtraction and the estimated delta of the raw log pitch. When both acoustic and prosodic features were used for input, we concatanated both features into one vector. In total 123-dimentional features were input in each time frame.

B. Results

Table IV shows the precision, recall, and F1 scores of fillers and word fragments when using different input features. We can see that the sequence labeling performance improved using acoustic features, and further improvement was also obtained with the prosodic features. These results indicate some fillers and word fragments could not be labeled using only lexical information. The acoustic and prosodic features can help the models to detect and label these disfluencies.

Table V shows the F1 scores of fillers and word fragments for different attention types. “No-attention” represents the model that used a uniform distribution for the attention weights. We evaluated content-based and location-based attention models using the F1 scores of fillers and word fragments. In this experiment, lexical, acoustic, and prosodic features were used for all models. The results were that all attention-based models provided improvement over the “No-attention” model. The attention mechanism appears to have worked effectively to emphasize speech features related to the disfluencies. Among them, location-based attention showed the best F1 score of both fillers and word fragments

V. Conclusion

This paper presented a method of neural network-based token-by-token sequence labeling for disfluency detection, one that which utilizes both speech and lexical information without any alignments between speech and text. We introduced attention mechanisms into the method for sequence labeling based on BLSTM-CRFs to handle both speech and lexical information. In our proposed method, the attention-based network emphasizes the speech features related to disfluencies. Experimental evaluations were conducted in sequence labeling task based on BIO labels of fillers and word fragments in Japanese spontaneous speech. In the experiments, the attention mechanism enabled improvement in the sequence labeling performance. We performed experiments with different input features: lexical, acoustic, and prosodic features. The best F1 scores were obtained when lexical, acoustic and prosodic features were utilized for input. Future work includes labeling the disfluencies to the ASR hypotheses and applying our proposed method to other disfluencies such as laughter and emphases.

TABLE IV
PRECISION, RECALL AND F1 SCORES OF FILLERS AND WORD FRAGMENTS WITH DIFFERENT MODELS.

Input features			Filler			Word fragmant		
Lexical	Acoustic	Prosodic	Precision	Recall	F1	Precision	Recall	F1
✓			94.76	91.69	93.20	54.07	67.08	59.87
✓	✓		95.23	91.81	93.49	53.98	69.80	60.88
✓		✓	95.11	91.80	93.43	54.77	69.63	61.31
✓	✓	✓	95.09	92.72	**93.89**	55.71	69.54	**61.86**

TABLE V
F1 SCORES OF FILLERS AND WORD FRAGMENTS WITH DIFFERENT ATTENTION TYPES. LEXICAL, ACOUSTIC AND PROSODIC FEATURES WERE USED FOR ALL MODELS.

Attention type	Filler	Word fragment
No-attention	93.11	59.88
Content-based-attention	92.70	61.64
Location-based attention	**93.89**	**61.86**

REFERENCES

[1] Y. Liu, E. Shriberg, A. Stolcke, D. Hillard, M. Ostendorf, and M. P. Harper, “Enriching speech recognition with automatic detection of sentence boundaries and disfluencies,” *IEEE Trans. Audio, Speech & Language Processing*, vol. 14, no. 5, pp. 1526–1540, 2006.

[2] V. Zayats, M. Ostendorf, and H. Hajishirzi, “Multi-domain disfluency and repair detection,” *In Proc. International Speech Communication Association (INTERSPEECH)*, pp. 2907–2911, 2014.

[3] J. Ferguson, G. Durrett, and D. Klein, “Disfluency detection with a semi-markov model and prosodic features,” *In Proc. North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 257–262, 2015.

[4] E. Cho, T.-L. Ha, and A. Waibel, “Crf-based disfluency detection using semantic features for german to english spoken language translation,” *In Proc of International Workshop for Spoken Language Translation (IWSLT)*, 2013.

[5] V. Zayats, M. Ostendorf, and H. Hajishirzi, “Disfluency detection using a bidirectional LSTM,” *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 2523–2527, 2016.

[6] S. Wang, W. Che, Y. Zhang, M. Zhang, and T. Liu, “Transition-based disfluency detection using lstms,” *In Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2785–2794, 2017.

[7] J. P. C. Chiu and E. Nichols, “Named entity recognition with bidirectional lstm-cnns,” *Transactions of the Association for Computational Linguistics (TACL)*, vol. 4, pp. 357–370, 2016.

[8] Z. Huang, W. Xu, and K. Yu, “Bidirectional LSTM-CRF models for sequence tagging,” *arXic prepring arXiv:1508.01991*, vol. abs/1508.01991, 2015.

[9] O. Klejch, P. Bell, and S. Renals, “Sequence-to-sequence models for punctuated transcription combining lexical and acoustic features,” *In Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5700–5704, 2017.

[10] Y. Wang, H. Huang, K. Chen, and H. Chen, “Discourse marker detection for hesitation events on mandarin conversation,” *In Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 1721–1725, 2018.

[11] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” *arXiv:1409.0473*, 2014.

[12] I. Sutskever, O. Vinyals, and Q. V. Le, “Sequence to sequence learning with neural networks,” *In Proc. Annual Conference on Neural Information Processing Systems (NIPS)*, pp. 3104–3112, 2014.

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

[14] S. Furui, K. Maekawa, and H. Isahara, “A Japanese national project on spontaneous speech corpus and processing technology,” *In Proc. ASR2000 - Automatic Speech Recognition: Challenges for the new Millenium*, pp. 244–248, 2000.

[15] M. D. Zeiler, “ADADELTA: an adaptive learning rate method,” *arXiv:1212.5701*, 2012.