INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

Joint prediction of punctuation and disfluency in speech transcripts

Binghuai Lin, Liyuan Wang

Smart Platform Product Department, Tencent Technology Co., Ltd, China
{binghuailin, sumerlywang}@tencent.com

Abstract

Spoken language transcripts generated from Automatic speech recognition (ASR) often contain a large portion of disfluency and lack punctuation symbols. Punctuation restoration and disfluency removal of the transcripts can facilitate downstream tasks such as machine translation, information extraction and syntactic analysis [1]. Various studies have shown the influence between these two tasks and thus performed modeling based on a multi-task learning (MTL) framework [2, 3], which learns general representations in the shared layers and separate representations in the task-specific layers. However, task dependencies are normally ignored in the task-specific layers. To model the dependencies of tasks, we propose an attention-based structure in the task-specific layers of the MTL framework incorporating the pretrained BERT (a state-of-art NLP-related model) [4]. Experimental results based on English IWSLT dataset and the Switchboard dataset show the proposed architecture outperforms the separate modeling methods as well as the traditional MTL methods.

Index Terms: Punctuation prediction, disfluency prediction, joint prediction, MTL, attention

1. Introduction

Speech transcripts from most ASR systems normally lack punctuation symbols and contain disfluencies. Due to these distinctive characteristics compared to the written language, it's essential to implement punctuation restoration and disfluency removal in spoken language processing [3].

Punctuation prediction is used for restoring punctuation symbols such as commas, periods and question marks of the unsegmented texts. Many studies have treated it as a sequence labeling task. Traditional methods based on conditional random fields (CRF) combine various kinds of textual features to predict punctuation marks [5, 6]. With the development of deep learning, approaches based on neural networks outperformed traditional methods by a wide margin. A deep neural network combined with CRF took prosodic features as input and generated sentence boundaries [7]. The convolution neural network (CNN) was developed to predict punctuation marks [8]. Other neural networks such as the recurrent neural network (RNN) have been widely used in the punctuation prediction task. The work in [9] presented a two-stage method based on a long short-term memory network (LSTM). A bidirectional RNN combined with the attention mechanism (T-BRNN) was proposed to restore punctuation and outperformed previous work [10]. A bidirectional LSTM with a CRF layer (BLSTM-CRF) and an ensemble model were proposed to improve the performance of the punctuation prediction [11]. Some other studies treated punctuation prediction as a machine translation problem, where the source is the unpunctuated text and the target is the text with punctuation. An RNN encoder-decoder structure with an attention layer was developed for punctuation restoration [12]. The work in [13] explored a self-attention based model to predict punctuation marks by combining both text and speech features and obtain state-of-the-art results in punctuation prediction.

Usually, disfluencies can be classified into two main types: filler words and edit words. The filler words include filled pauses (e.g., 'uh', 'um') and discourse marks (e.g., 'you know', 'i mean'). The edit words usually mean words that are spoken wrongly and corrected by the speaker. Many approaches have been proposed for disfluency prediction. Methods based on CRF were employed to detect disfluencies in spoken language transcripts [14, 15]. A BLSTM neural network was introduced for disfluency detection [16]. A convolution neural network with an auto-correlation operator was developed for disfluency detection [17]. A semi-supervised approach based on self-attention mechanism was proposed to predict disfluency [18]. The work in [19] adapted neural machine translation model (NMT) on the basis of transformer [20] and outperformed previous works. Recently, the BLSTM-based model combining residual BLSTM blocks, self-attention, and a noisy training approach was introduced in [21] and achieved a strong performance.

These models focus on improving either the punctuation prediction accuracy or the disfluency detection accuracy. It has been found that there is mutual influence between punctuation restoration and disfluency prediction in [22]. It proposed three combined methods based on the CRF model and found these methods outperformed the isolated prediction method by 0.5%-1.5% based on F1-measure. A two-stage approach was proposed in [23] to do sentence segmentation followed by simple-disfluency removal first and then do complex-disfluency removal. With the development of neural network, the methods based on an MTL framework, which learns general feature representations for all the tasks by parameter sharing and task-specific representations in the task-specific layers, were proposed for joint modeling of two tasks [2, 3]. However, the dependencies of these two tasks were ignored in the task-specific layers.

Recently, attention mechanism has been used in various problems like image captioning [24], neural machine translation [20] and automatic speech recognition [25]. Attention mechanism can have access to the global sequence features and place more attention on the relevant features. The contextual influence of punctuation prediction (disfluency detection) on disfluency detection (punctuation prediction) can be local or global. In this paper, we propose an attention-based structure in the task-specific layers to model the dependencies of disfluency detection and punctuation prediction based on an MTL framework. As BERT has advanced the state-of-the-art in various NLP tasks [4], we model the combination of punctuation prediction and disfluency detection based on this particular network. In section 2, we will introduce the proposed joint methods. The experiments are conducted in section 3. We will draw the conclusions and future suggestions in section 4.

http://dx.doi.org/10.21437/Interspeech.2020-1277

2. Proposed method

2.1. Modeling

In this paper, we consider punctuation prediction and disfluency detection as sequence labeling tasks. Our baseline is based on BERT. The architecture of BERT is a multi-layer bidirectional Transformer encoder based on the original Transformer model [26]. The input of BERT is a sequence of word pieces (subword units) [27] in the sentence $x = (x_1, x_2...x_n)$ and the output is $H = (h_1, h_2...h_n)$. Given the final output H of BERT, a sequence of results $p = (p_1, p_2...p_n)$ where p_i represents punctuation (e.g., comma (C), period (P), question (Q), blank mark (B)) or $d = (d_1, d_2...d_n)$ where d_i represents disfluency results (e.g., disfluency, fluency) is predicted based on a fully connected layer (FC) defined in Eq. (1). The results of punctuation prediction and disfluency detection can be achieved as Eq. (1):

$$p_i = \text{softmax}(w \times h_i + b) \tag{1}$$

where w and b are the trainable weights and biases in the network.

Traditional methods based on the MTL framework share general representations H and learns specific representations in the task-specific layers. Thus, prediction of punctuation and disfluency based on the MTL framework can be formulated as joint probability of punctuation and disfluency, making the hypothesis that two tasks are independent of each other conditioned on h. Specifically,

$$P(p, d|h) = P(p|h) \times P(d|h) \tag{2}$$

Normally, $P(p|h)$ and $P(d|h)$ are modeled in separate task-specific FC layers as in Eq. (1).

Besides general features shared by two tasks, there is mutual influence between punctuation and disfluency results. For example, 'I like' might be predicted as repetition disfluency in the sentence 'I like football I like basketball', while it might not in the sentence 'I like football. I like basketball'. Assuming the two tasks are not independent of each other, joint probability of punctuation and disfluency can be formulated by conditional probability. Specifically, predicting punctuation conditioned on disfluency results is shown in Eq. (3) and predicting disfluency conditioned on punctuation results is defined in Eq. (4):

$$P(p, d|h) = P(p|d, h) \times P(d|h) \tag{3}$$

$$P(p, d|h) = P(d|p, h) \times P(p|h) \tag{4}$$

That is, instead of modeling conditional independent $P(p|h)$ and $P(d|h)$ as in Eq. (2), we will model $P(p|d, h)$ and $P(d|h)$ or $P(d|p, h)$ and $P(p|h)$ directly using particular network structures. The traditional MTL method is shown in Figure 1 and the proposed MTL considering task dependencies is displayed in Figure 2.

The most common principle by which a model is fitted to the data is by the maximum likelihood (ML) principle [28], the likelihood can be represented as Eq. (5):

$$\begin{aligned}\text{Likelihood} &= \prod_{i=0}^{n}\prod_{k=1}^{c} P(p, d|h)^{y(p,d)} \\ &= \prod_{i=0}^{n}\prod_{k=1}^{c} P(p|h)^{y(p)} P(d|p, h)^{y(d)} \\ &= \prod_{i=0}^{n}\prod_{k=1}^{c} P(p|d, h)^{y(p)} P(d|h)^{y(d)}\end{aligned} \tag{5}$$

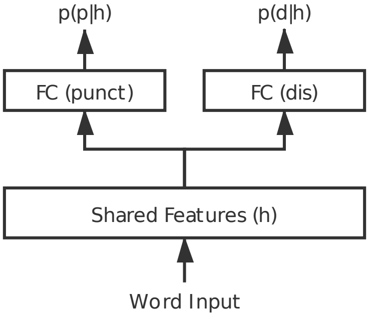

Figure 1: *Traditional MTL*

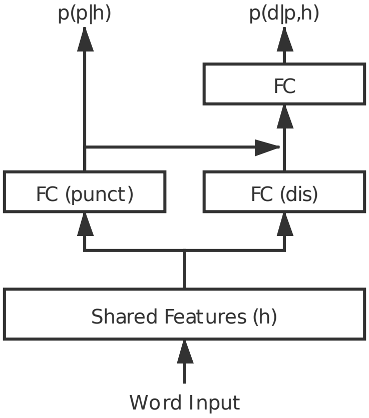

Figure 2: *Proposed MTL*

where c represents the number of categories of combined punctuation and disfluency and n represents the total number of samples. y is the ground truth of punctuation or disfluency and p is the probability derived from the Eq. (1).

The maximization of the likelihood can be formulated as the minimization of negative log likelihood. The final loss function for joint optimization of these two tasks can be defined in Eq. (6):

$$\begin{aligned}L_{\text{total}} &= -\sum_{i=1}^{n} y(p, d) \times \log P(p, d|h) \\ &= -\sum_{i=1}^{n} y(p) \times \log P(p|h) - \sum_{i=1}^{n} y(d) \times \log P(d|p, h) \\ &= -\sum_{i=1}^{n} y(p) \times \log P(p|d, h) - \sum_{i=1}^{n} y(d) \times \log P(d|h) \\ &= L_{\text{punctuation}} + L_{\text{disfluency}}\end{aligned} \tag{6}$$

where $L_{\text{punctuation}}$ and $L_{\text{disfluency}}$ are the classification losses of punctuation and disfluency prediction. Usually, a value ranging from 0 to 1 is set to balance the weight between two tasks for fast convergence of the algorithm.

2.2. Network architecture

The joint probability of two tasks with dependencies mentioned in the previous section can be implemented by two methods: (1) predicting disfluency based on punctuation results; (2) predicting punctuation based on disfluency results. The dependencies of two tasks are implemented by an attention based structure in the task-specific layers shown in Figure 3. The attention function can be described as mapping a query and a set of key-value pairs to an output, which was proposed in neural machine translation [29]. Specifically, to predict disfluency based on punctuation results, the punctuation probability results $p = (p_1, p_2 ... p_n)$ and disfluency results $d = (d_1, d_2 ... d_n)$ are achieved from two separate FC layers. The transformed predicted punctuation results can be treated as the keys and values, and each disfluency result d_j can be taken as the query. The final punctuation feature f_j can be achieved by the attention mechanism defined in Eq. (7):

$$f_j = \sum_{i \in n} \alpha_i pc_i \tag{7}$$

where α_i is the attention weight defined in Eq. (8) and n is the number of samples (words) in a sentence.

$$\alpha_i = \frac{\exp(pc_i^T d_j)}{\sum_{k \in n} \exp(pc_k^T d_j)} \tag{8}$$

where pc_i are transformed predicted probability of punctuation shown in Eq. (9):

$$pc_i = \tanh(w_{pc} * p_i + b_{pc}) \tag{9}$$

The punctuation features f_j and d_j are concatenated together followed by an FC layer to predict final disfluency results. Predicting punctuation based on disfluency results can be implemented similarly. The final disfluency results are shown in Eq. (10):

$$d_j(final) = \mathrm{softmax}(w_c \times [f_j, d_j] + b_c) \tag{10}$$

where $[f_j, d_j]$ are the operation of concatenation of f_j and d_j. w_c and b_c are the trainable weights and bias for feature concatenation which are randomly initialized in the network.

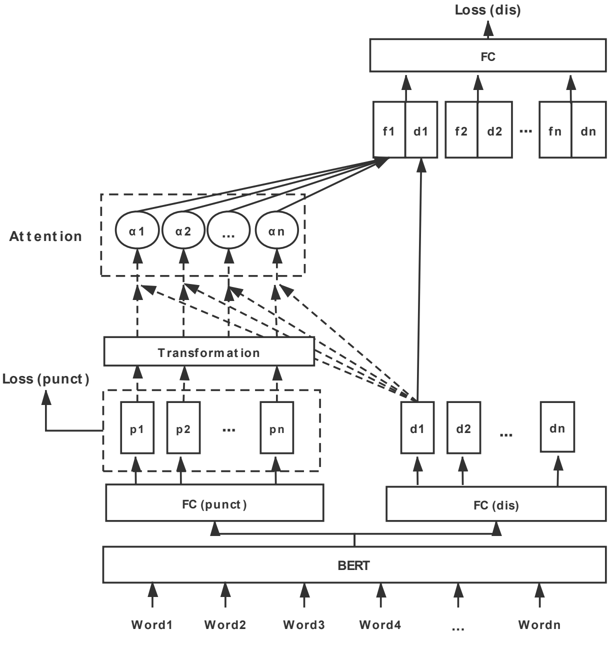

Figure 3: *Proposed structure based on attention*

3. Experiments

3.1. Corpus

We assess the proposed method for punctuation prediction and disfluency detection on the Switchboard corpus of conversational speech [30] and English IWSLT dataset. Following previous work in [16, 17], we split the Switchboard corpus into training, dev and test set as follows: training data consists of all sw[23].dff files, dev training consists of all sw4[5-9].dff files, and test data consists of all sw4[0-1].dff files. In English IWSLT dataset, there are three datasets: training set, development set and test set. The training set and development set are from the training data of IWSLT2012 machine translation track. The training set contains about 2.1M words, 144K sentences. The development set has about 296K words, 21K sentences. There are two test sets: reference and ASR, which are from the IWSLT2011. The test set contains about 13K words, 860 sentences. We use these training, development and test sets to train and test our models as previous work did [10, 11, 13].

3.2. Baseline results

We obtain the isolated baselines for punctuation and disfluency prediction based on two separate BERT models. For comparing different combination methods, we present two baselines for punctuation which are based on the IWSLT dataset and the Switchboard dataset separately. The disfluency detection baseline is based on the Switchboard dataset. As the utterances of each speaker have been segmented into short sentences in the Switchboard dataset, we join utterances into long sentences for punctuation evaluation. We use the English uncased BERT-Base model, which has 12 layers, 768 hidden states, and 12 heads. The BERT fine-tuned model is trained with epochs ranging from 5 to 8. All models are evaluated in terms of punctuation and disfluency precision, recall and F1-score.

From the results of the punctuation prediction shown in Table 1, we can see that the BERT fine-tuned baseline used in our experiments achieved the state-of-art results in most testsets. The performance of disfluency detection trained on the Switchboard dataset is evaluated from two perspectives: (1) short sentences; (2) long sentences joining from short utterances. The results of disfluency detection in Table 3 show the BERT fine-tuned baseline outperforms the previous work by 2% in F1-score. The performance based on assembled long sentences is inferior to that on short sentences by 9% in F1-measure, indicating potential influence of sentence segmentation.

3.3. Joint approach

We implement combination of the punctuation prediction and disfluency detection based on two strategies: (1) combining the punctuation task based the IWSLT dataset and the disfluency task on the Switchboard dataset (SEP); (2) combining punctuation and disfluency tasks both based on the Switchboard dataset (COMBINE (SWBD)). We compare results of different combinations based on the traditional MTL method (feature sharing (FS)) and our proposed approaches, including predicting punctuation after disfluency (DP) and predicting disfluency after punctuation (PD).

From the results shown in Table 2, we can see that there is 1% improvement on average for punctuation of IWSLT REF dataset in SEP and nearly 1% improvement for the Switchboard dataset in COMBINE (SWBD), while there is little improvement for the traditional MTL method (FS), indicating superiority of modeling dependencies of two tasks based on the

Table 1: *Punctuation comparison results of the baseline*

TRAIN	TEST	MODEL	COMMA			PERIOD			QUESTION		
			P	R	F1	P	R	F1	P	R	F1
IWSLT	ASR	T-BRNN-PRE [10]	59.6	42.9	49.9	70.7	72	71.4	60.7	48.6	54.0
		Teacher-Ensemble [11]	60.6	58.3	59.4	71.7	72.9	72.3	66.2	55.8	60.6
		Self-attention word [13]	**61.5**	57.2	**59.3**	72.1	73.0	72.5	**67.9**	60.6	**64.0**
		BERT finetune	54.2	**63.1**	58.3	**75.3**	**76.2**	**75.7**	57.6	**63.0**	61.7
	REF	T-BRNN-PRE [10]	65.5	47.1	54.8	73.3	72.5	72.9	70.7	63.0	66.7
		Teacher-Ensemble [11]	66.2	59.9	62.9	75.1	73.7	74.4	72.3	63.8	67.8
		Self-attention word [13]	64.9	58.7	61.6	79.1	74.6	76.8	75.4	64.9	69.8
		BERT finetune	**68.2**	**68.8**	**68.5**	**81.2**	**81.3**	**81.2**	**81.2**	**81.3**	**82.1**
SWBD	SWBD TEST	Recurrent network [2]	80.0	76.1	78.0	78.2	64.2	70.5	78.0	58.2	66.7
		BERT finetune	**80.2**	**85.0**	**82.5**	**91.2**	**77.1**	**83.6**	**80.2**	**65.8**	**72.3**

Table 2: *Punctuation results of combinations*

	COMBINE (SEP)			COMBINE (SWBD)		
	F1 (C)	F1 (P)	F1 (Q)	F1 (C)	F1 (P)	F1 (Q)
IWSLT ASR(BASE)	58.3	75.7	61.7	39.2	69.8	45.7
IWSLT ASR(FS)	58.5	75.5	61.2	38.9	70.1	44.1
IWSLT ASR(DP)	58.1	75.3	61.5	**43.2**	71.2	47.3
IWSLT ASR(PD)	58.5	75.8	61.9	41.3	**71.5**	**51.2**
IWSLT REF(BASE)	68.5	81.2	82.1	51.5	76.2	53.1
IWSLT REF(FS)	69.1	80.2	81.9	51.6	76.1	52.3
IWSLT REF(DP)	69.8	**82.3**	**83.1**	51.8	76.2	58.1
IWSLT REF(PD)	**70.1**	82.1	81.8	**52.1**	**76.5**	**58.3**
SWBD (BASE)	61.3	57.5	44.3	82.5	83.6	72.3
SWBD (FS)	61.5	58.4	48.5	82.1	83.5	72.5
SWBD (DP)	61.7	**60.1**	49.8	82.2	83.1	72.1
SWBD (PD)	**62.4**	59.2	**51.5**	**83.4**	**84.5**	**73.1**

Table 3: *Disfluency results of the baseline*

	P	R	F1
Weight sharing [18]	92.1	90.2	91.1
BLSTM [16]	91.6	80.3	85.9
Translation-based [19]	94.5	84.1	89.0
EGBC [21]	95.7	88.3	91.8
Short sents (BERT)	**96.1**	**91.7**	**93.8**
Long sents (BERT)	82.6	86.1	84.3

Table 4: *Disfluency results of combinations*

	F1(Short)	F1(Long)
BASE	93.8	84.3
FS(SEP)	93.6	74.2
DP(SEP)	93.5	75.5
PD(SEP)	94.1	74.1
FS(SWBD)	93.5	87.1
DP(SWBD)	93.7	88.5
PD(SWBD)	**94.3**	**89.2**

MTL framework over the traditional MTL method. The performance of punctuation based on the Switchboard dataset improves nearly 1% in comma prediction, 2% in period prediction, and 5% in question mark prediction in COMBINE (SEP) / SWBD (PD/DP). Despite of lacking Switchboard punctuation labels in SEP training data, the results demonstrate that the disfluency task can facilitate the generalization of punctuation prediction by sharing general features in the shared layers. The same improvement in IWSLT dataset can be observed in COMBINE(SWBD) / IWSLT (PD/DP).

The disfluency results are shown in Table 4. From the results, we can see there is little improvement in the disfluency detection based on the short sentences while there exists obvious improvement or degradation based on the long sentences. The disfluency detection performance degrades nearly 10% in the separate dataset (DP(SEP) / PD(SEP)), indicating disfluency detection of the long sentences is greatly influenced by the accuracy of punctuation prediction. The performance improves nearly 3% by combining two tasks with true punctuations as shown in FS (SWBD). By modeling dependencies of two tasks, there is additional 2% improvement shown in PD (SWBD) compared with the traditional MTL method.

4. Conclusions

In this paper, we propose an attention based structure in the MTL framework to model the dependencies of punctuation and disfluency prediction tasks. Experimental results based on the IWSLT and Switchboard datasets show the combinations can improve the performance and generalization of two tasks compared to a strong baseline from the separate modeling and the tradition MTL modeling. As disfluency and punctuation are related to prosodic features, we will investigate combining both textual and prosodic features based on the proposed method in the future.

5. References

[1] S. Peitz, M. Freitag, A. Mauser, and H. Ney, "Modeling punctuation prediction as machine translation," in *International Workshop on Spoken Language Translation (IWSLT) 2011*, 2011.

[2] M. Reisser, "Recurrent neural networks in speech disfluency detection and punctuation prediction," 2015.

[3] E. Cho, K. Kilgour, J. Niehues, and A. Waibel, "Combination of nn and crf models for joint detection of punctuation and disfluencies," in *Sixteenth annual conference of the international speech communication association*, 2015.

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[5] W. Lu and H. T. Ng, "Better punctuation prediction with dynamic conditional random fields," in *Proceedings of the 2010 conference on empirical methods in natural language processing*, 2010, pp. 177–186.

[6] N. Ueffing, M. Bisani, and P. Vozila, "Improved models for automatic punctuation prediction for spoken and written text." in *Interspeech*, 2013, pp. 3097–3101.

[7] C. Xu, L. Xie, G. Huang, X. Xiao, E. S. Chng, and H. Li, "A deep neural network approach for sentence boundary detection in broadcast news," in *Fifteenth annual conference of the international speech communication association*, 2014.

[8] X. Che, C. Wang, H. Yang, and C. Meinel, "Punctuation prediction for unsegmented transcript based on word vector," in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, 2016, pp. 654–658.

[9] O. Tilk and T. Alumäe, "Lstm for punctuation restoration in speech transcripts," in *Sixteenth annual conference of the international speech communication association*, 2015.

[10] ——, "Bidirectional recurrent neural network with attention mechanism for punctuation restoration." in *Interspeech*, 2016, pp. 3047–3051.

[11] J. Yi, J. Tao, Z. Wen, Y. Li *et al.*, "Distilling knowledge from an ensemble of models for punctuation prediction," 2017.

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

[13] J. Yi and J. Tao, "Self-attention based model for punctuation prediction using word and speech embeddings," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 7270–7274.

[14] J. Ferguson, G. Durrett, and D. Klein, "Disfluency detection with a semi-markov model and prosodic features," in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2015, pp. 257–262.

[15] E. Fitzgerald, K. B. Hall, and F. Jelinek, "Reconstructing false start errors in spontaneous speech text," 2009.

[16] V. Zayats, M. Ostendorf, and H. Hajishirzi, "Disfluency detection using a bidirectional lstm," *arXiv preprint arXiv:1604.03209*, 2016.

[17] P. J. Lou, P. Anderson, and M. Johnson, "Disfluency detection using auto-correlational neural networks," *arXiv preprint arXiv:1808.09092*, 2018.

[18] F. Wang, W. Chen, Z. Yang, Q. Dong, S. Xu, and B. Xu, "Semi-supervised disfluency detection," in *Proceedings of the 27th International Conference on Computational Linguistics*, 2018, pp. 3529–3538.

[19] Q. Dong, F. Wang, Z. Yang, W. Chen, S. Xu, and B. Xu, "Adapting translation models for transcript disfluency detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 6351–6358.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[21] N. Bach and F. Huang, "Noisy bilstm-based models for disfluency detection," *Proc. Interspeech 2019*, pp. 4230–4234, 2019.

[22] X. Wang, H. T. Ng, and K. C. Sim, "Combining punctuation and disfluency prediction: an empirical study," 2014.

[23] H. Hassan, L. Schwartz, D. Hakkani-Tür, and G. Tur, "Segmentation and disfluency removal for conversational speech translation," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[24] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International conference on machine learning*, 2015, pp. 2048–2057.

[25] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Advances in neural information processing systems*, 2015, pp. 577–585.

[26] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[27] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," *arXiv preprint arXiv:1508.07909*, 2015.

[28] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

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

[30] J. J. Godfrey, E. C. Holliman, and J. McDaniel, "Switchboard: Telephone speech corpus for research and development," in *[Proceedings] ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1. IEEE, 1992, pp. 517–520.