Interspeech 2022
18-22 September 2022, Incheon, Korea

Bring dialogue-context into RNN-T for streaming ASR

Junfeng Hou, Jinkun Chen*, Wanyu Li, Yufeng Tang, Jun Zhang, Zejun Ma*

Speech & Audio Team, AI Lab, ByteDance Inc., Beijing, China

{houjunfeng, chenjinkun.kv7}@bytedance.com

Abstract

Recently the conversational end-to-end (E2E) automatic speech recognition (ASR) models, which directly integrate dialogue-context such as historical utterances into E2E models, have shown superior performance than single-utterance E2E models. However, few works investigate how to inject the dialogue-context into the recurrent neural network transducer (RNN-T) model. In this work, we bring dialogue-context into a streaming RNN-T model and explore various structures of contextual RNN-T model as well as training strategies to better utilize the dialogue-context. Firstly, we propose a deep fusion architecture which efficiently integrates the dialogue-context within the encoder and predictor of RNN-T. Secondly, we propose contextual & non-contextual model joint training as regularization, and propose context perturbation to relieve the context mismatch between training and inference. Moreover, we adopt a context-aware language model (CLM) for contextual RNN-T decoding to take full advantage of the dialogue-context for conversational ASR. We conduct experiments on the Switchboard-2000h task and observe performance gains from the proposed techniques. Compared with non-contextual RNN-T, our contextual RNN-T model yields 4.8% / 6.0% relative improvement on Switchboard and Callhome Hub5'00 testsets. By additionally integrating a CLM, the gain is further increased to 10.6% / 7.8%.

Index Terms: speech recognition, streaming RNN-T, dialogue-context aware

1. Introduction

End-to-end (E2E) automatic speech recognition (ASR) approaches such as connectionist temporal classification (CTC) [1], recurrent neural network transducer (RNN-T) [2, 3] and listen-attend-spell (LAS) [4] have achieved superior performance over the hybrid HMM-DNN systems [5] when sufficient training data is available. By integrating the acoustic model (AM), pronunciation model (PM) and language model (LM) within a single neural network, the training and inference pipeline of these E2E ASR models are greatly simplified. Moreover, benefiting from the E2E modeling property, various contextual information can be incorporated into the E2E model [6, 7, 8] to improve performance.

In contextual-LAS (CLAS) [6, 9] and context-aware transformer transducer (CATT) [7], contextual phrases are dynamically processed by an attentive module to recognize rare words. Similar strategies are adopted in contextual RNN-T [10, 11] for generating video subtitles and keyword spotting. Other contexts such as dialogue state [12], intent [13] and location [14] are also proven to be helpful for E2E models. When it comes to conversational ASR, a long-range dialogue-context consisting of historical utterances or previous turns in a dialogue has been introduced to E2E models. The global conversational-level consistency such as conversation topic and tendency occurs across utterances in a dialogue [15]. Therefore modeling such dialogue-context makes an utterance more recognizable. Dialogue-context is firstly introduced in LAS to process long conversations [16]. External semantic embedding from BERT [17] and gating mechanism [18] further improve the ASR performance. In a conversation scenario, streaming RNN-T is preferable for recognizing conversational speech in real time. Therefore, dialogue-context is introduced to RNN-T for speech recognition [19]. Additionally, LM is also allowed to operate across utterance boundaries [20], and contextual LM is proposed to selectively leverage the contextual information[15, 21].

Although injecting dialogue-context into RNN-T predictor has improved the performance [19], it is unclear which RNN-T's component is most suitable for integrating dialogue-context: the encoder, the predictor or both. Moreover, it is crucial to design better dialogue-context aware architectures and training strategies for RNN-T to better utilize the context, especially considering the diverse relationships between utterances. In addition, since the dialogue-context is usually ground-truth historical utterances during training and model hypotheses during inference, the contextual model may suffer from the dialogue-context mismatch between training and inference stages. While an utterance-level sampling strategy is used in [18], we argue that generating hypotheses during training is time-consuming. What's more, in practical applications, if the dialogue-context is not available, this context missing issue is expected not to degrade the performance of the context-aware RNN-T model.

In this paper, we first inject the dialogue-context into a streaming RNN-T's encoder, predictor, or both, and compare the three contextual models' performance on the Switchboard-2000h task. Different from the CATT model [7] where injecting contextual phrases into both encoder and predictor significantly outperforms injecting contextual phrases into encoder only, we find that injecting dialogue-context into encoder is slightly better than predictor and no more considerable gain is found when injecting dialogue-context into both encoder and predictor. Furthermore, instead of integrating context into RNN-T by adding a contextual module after the encoder or predictor [7], we propose a deep fusion architecture which integrates the dialogue-context within the encoder and predictor to combine the context more efficiently. We also propose contextual & non-contextual model joint training and context perturbation for contextual model optimization. We show that the joint training not only improves the contextual performance, but also alleviates the context missing issue. And the context perturbation eliminates the context mismatch between training and inference. Moreover, we observe further performance gain from integrating contextual LM for contextual RNN-T decoding.

The rest of this paper is organized as follows: Section 2 briefly introduces the frameworks of RNN-T system, then describes the proposed context-aware RNN-T models and train-

* Equal contribution

10.21437/Interspeech.2022-697

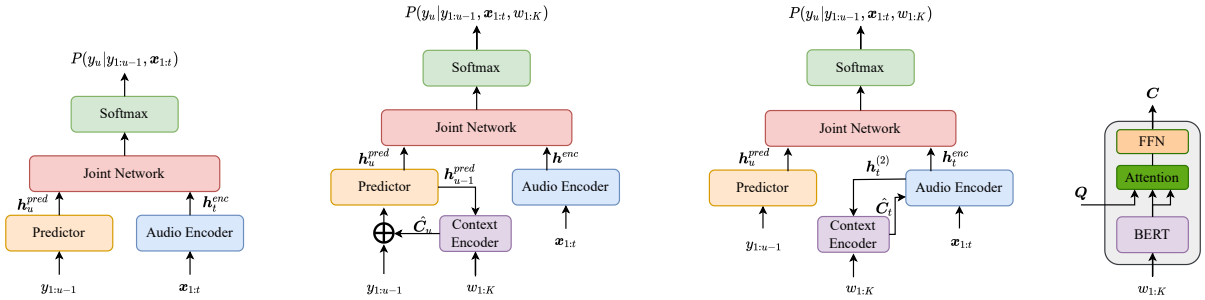

Figure 1: *(a) Standard RNN-T, (b) RNN-T with CAP, (c) RNN-T with CAE, (d) attentive context encoder.*

ing strategies. Section 3 presents the details of datasets, experiments and discussions, followed by conclusions in Section 4.

2. Methods

2.1. RNN-T

Figure 1(a) illustrates the framework of a standard RNN-T model, which consists of an encoder, a predictor and a joint network. Given the T-frames acoustic features $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_T]$ and its corresponding label sequence $\boldsymbol{Y} = [y_1, y_2, ..., y_U]$, the encoder network converts $\boldsymbol{X}$ to a sequence of high-level representations $\boldsymbol{H}^{enc} = [\boldsymbol{h}_1^{enc}, \boldsymbol{h}_2^{enc}, ..., \boldsymbol{h}_{T'}^{enc}]$ with down sampling size of D, where $T' = T/D$,

$$\boldsymbol{H}^{enc} = f^{enc}(\boldsymbol{X}). \tag{1}$$

The predictor network encodes non-blank historical tokens $y_{1:u-1}$ to label representation $\boldsymbol{h}_u^{pred}$ by

$$\boldsymbol{h}_u^{pred} = f^{pred}(y_{1:u-1}). \tag{2}$$

And then, the joint network combines $\boldsymbol{h}_t^{enc}$ and $\boldsymbol{h}_u^{pred}$ with a feed-forward network followed by a softmax function to calculate the probability distribution $p(y|t,u)$ over output labels plus a blank symbol at frame t and step u,

$$p(y|t,u) = \text{Softmax}(f^{joint}(\boldsymbol{h}_t^{enc}, \boldsymbol{h}_u^{pred})). \tag{3}$$

With forward-backward algorithm [2], the probability of output $\boldsymbol{Y}$ given input $\boldsymbol{X}$ can be calculated for RNN-T training.

2.2. Context encoder

As shown in Figure 1(d), we adopt BERT to convert the word sequence of dialogue-context $[w_1, w_2, ..., w_K]$ to semantic embedding $\boldsymbol{C}_{embd}$, and stack an multi-head attention layer to attend the contextual embedding by query $\boldsymbol{Q}$, where $\boldsymbol{Q}$ is the label or the audio representations. Afterwards, a feed-forward network (FFN), which consists of two linear layers with a relu activation in between, projects the attention layer's output $\boldsymbol{C}$ to proper dimension. These calculations are formulated as

$$\boldsymbol{C}_{embd} = \text{BERT}([w_1, w_2, ..., w_K]), \tag{4}$$

$$\boldsymbol{C} = \text{FFN}(f^{attn}(\boldsymbol{Q}, \boldsymbol{C}_{embd})), \tag{5}$$

where f^{attn} denotes the standard attention [22] function which takes $\boldsymbol{Q}$ as *query*, while $\boldsymbol{C}_{embd}$ as *key* and *value*. Before injecting context $\boldsymbol{C}$ into RNN-T model, gating mechanism [18] is adopted to downscale the contribution of the context when needed. The gated output $\hat{\boldsymbol{C}}$ is calculated by

$$\boldsymbol{g} = \text{Sigmoid}(\boldsymbol{W}(\boldsymbol{Q}, \boldsymbol{C}) + \boldsymbol{b}), \tag{6}$$

$$\hat{\boldsymbol{C}} = \boldsymbol{g} \odot \boldsymbol{C}, \tag{7}$$

where $\boldsymbol{g}$ is the gating value.

This attentive context encoder is integrated into the RNN-T model with context-aware predictor (section 2.3.1) and encoder (section 2.3.2) respectively.

2.3. Context-aware RNN-T

We explore different designs for incorporating dialogue-context into RNN-T. The context can be injected into RNN-T from the predictor, the audio encoder, or both simultaneously. Note that our model encodes the label or audio input together with the dialogue-context in the early stage to produce more efficiently fused representations, which is different from previous work.

2.3.1. Context-aware predictor (CAP)

Figure 1(b) illustrates the framework of RNN-T with context-aware predictor (CAP). For the LSTMs predictor network, let $\boldsymbol{e}_{u-1}$ and $\boldsymbol{h}_{u-1}^{pred}$ denote the word embedding and the LSTMs' output of the previous token respectively, then the gated context output $\hat{\boldsymbol{C}}_u$ is calculated by equation (7) with $\boldsymbol{h}_{u-1}^{pred}$ as the attention *query* $\boldsymbol{Q}$. Next, add $\hat{\boldsymbol{C}}_u$ and $\boldsymbol{e}_{u-1}$ together to feed into the predictor by

$$\boldsymbol{h}_u^{pred} = \text{LSTMs}(\boldsymbol{e}_{u-1} + \hat{\boldsymbol{C}}_u, \boldsymbol{h}_{u-1}^{pred}). \tag{8}$$

2.3.2. Context-aware encoder (CAE)

For the RNN-T encoder, we stack 8 uni-directional LSTM layers with time-reduced layer after the first two LSTM layers. The time-reduced layers down-sample the acoustic feature sequence along the frames. The design of RNN-T with context-aware encoder (CAE) is shown in 1(c). We take the down-sampled output of the second LSTM layer $\boldsymbol{h}_t^{(2)}$ as the attention *query* $\boldsymbol{Q}$ in equation (5). And $\boldsymbol{h}_t^{(2)}$ is combined with gated context output $\hat{\boldsymbol{C}}_t$, which is fed to the following 6 LSTM layers to produce the output $\boldsymbol{h}_t^{enc}$. These calculations are formulated as

$$\boldsymbol{h}_t^{(2)} = \boldsymbol{h}_t^{(2)} + \hat{\boldsymbol{C}}_t, \tag{9}$$

$$\boldsymbol{h}_t^{enc} = \text{LSTMs}(\boldsymbol{h}_t^{(2)}, \boldsymbol{h}_{t-1}^{enc}). \tag{10}$$

We place the attentive context encoder right after the downsampling layer since it significantly reduces the computation cost and facilitates the efficient integration between the dialogue-context and acoustic features.

2.3.3. Context-aware predictor & encoder (CAP & CAE)

Motivated by [7], both the encoder and predictor of RNN-T can integrate dialogue-context simultaneously, thus two attentive context encoders with different *query* are needed. The cal-

culation procedure is just a combination of CAP and CAE, and the details are omitted for simplicity.

2.4. Training Strategies

2.4.1. Joint training

Similar to [18], we initialize the context-aware E2E model from a baseline E2E model. Empirically we find that joint training the context-aware model and non-contextual model leads to better performance thus is adopted in all of our experiments. The loss weight of each model is set to 0.5. Meanwhile, this strategy relieves the possible performance degradation during inference when the dialogue-context is missing.

2.4.2. Context perturbation

In general, the ground truth transcripts of dialogue histories are used as context (named oracle context) in the training stage. However, ASR hypotheses are the dialogue-context during inference which makes the model suffer from recognition errors in the previous dialogue turns. To this end, Kim [18] applied schedule sampling [23] with a ratio of 20% to generate inferred context during training stage, which would increase the computation cost strikingly. Instead, we perturb the dialogue-context with probability 0.1 to randomly substitute, insert or delete tokens on-the-fly. This method efficiently relieves the context discrepancy between training and inference stages.

3. Experiments

3.1. Datasets

The experiments are conducted on the Switchboard-2000h task, which has around 2000 hours of conversational telephone speech from Switchboard-300 [24, 25] and Fisher [26, 27] corpus. We prepare the training, validation and test sets with the recipes in *ESPnet* [28]. The test set includes Switchboard (SWB) / CallHome (CH) subsets of the NIST Hub5'000 evaluation. The Table 1 shows details of the datasets.

Table 1: *Experimental dataset description.*

Dataset	# of utter.	# of conversations
training	2,141,808	28,236
validation	4,000	34
eval. (SWB)	1,831	20
eval. (CH)	2,627	20

3.2. Experimental setup

The streaming RNN-T baseline has the following configurations. For the RNN-T's encoder, we stack 8 uni-directional LSTM layers with a time-reduced layer after the first two LSTM layers. The LSTM is of hidden size 1024 and projects outputs to the dimension of 512. Each of the two time-reduced layers unfolds the LSTM outputs with stride of 2 along the dimension of frames, from which the acoustic features are downsampled with a factor of 4. For the RNN-T's predictor, 2-layer LSTM without projection layer is adopted. The RNN-T's joint network is a feed-forward layer. The context-aware RNN-T model has the following configurations. We explore the 12-layer, 110M BERT-Base model [29] as the context encoder, which is fixed during training. The hidden size of context embedding output $\hat{\boldsymbol{C}}_u$ and $\hat{\boldsymbol{C}}_t$ are 640 and 512 respectively. For labels, we use 2k byte-pair encoding (BPE) [30] sub-word units plus BLANK symbol.

Table 2: *The WERs of different context-aware RNN-T models*

Model	**Integration Type**	**SWB**	**CH**
baseline RNN-T	-	10.4	16.6
Prior Models*			
CAP [19]	shallow	10.2	16.1
CAE [7]		10.2	16.0
CAP & CAE [7]		10.2	16.0
Proposed Models			
CAP	deep	10.1	16.0
CAE		10.0	16.0
CAP & CAE		**10.0**	**15.9**

* We implement the same structures of prior contextual RNN-T models with dialogue-context as input.

Besides, a neural network language model (NNLM) consisting of 2 LSTM layers of hidden size 1024 is trained using the same sub-word units on training speech transcripts of the fisher corpus. We also use a contextual LM (CLM) to leverage dialogue-context for shallow fusion[31] in decoding. The CLM has the same architecture as the CAP but a different hidden size.

3.3. Results and discussion

3.3.1. Comparison of different context-aware RNN-T models

We firstly present the WERs of different context-aware RNN-T with single turn dialogue-context in the Table 2. All contextual models are initialized from the baseline RNN-T and no extra training strategies are adopted here. For ease of explanation, we refer to the prior models [7, 19] as the CAP, the CAE, or the CAP & CAE based on which RNN-T's component the dialogue-context is injected into. And we refer to the context integration type in prior models as "shallow integration" to distinguish our proposed deep integration architecture.

For both the CAP, the CAE and the CAP & CAE models, the proposed deep integration architecture outperforms shallow integration in prior models and achieves at most 3.8% / 4.2% relative gain over the baseline RNN-T on the SWB and CH subset. The CAE model is slightly better than CAP model, which can be explained by the findings in [32] that the RNN-T encoder and joint networks capture both the acoustic and linguistic information. While the CAP & CAE model yields the best performance, its performance is comparable with CAE model. To avoid the high computation cost of the CAP & CAE model, we use the CAE model with deep integration for the following experiments.

3.3.2. Evaluations of the proposed training strategies

We evaluate the effectiveness of the joint training strategy for the CAE model. The results in the Table 3 show that joint train-

Table 3: *The WERs of CAE without / with joint training*

model	**context**	**SWB**	**CH**
baseline RNN-T	-	10.4	16.6
CAE	hyp	10.0	16.0
	missing	10.3	16.4
+ joint training	hyp	10.1	**15.7**
	missing	10.2	16.2

Table 4: *Examples of reference, hypothesis from baseline, and our CAE model selected from the CH testset.*

Reference	Baseline	CAE
.. you use a higher **voltage** .. **burn a chip** .. a **programmer** which has a special **socket** that	.. you use a higher **voltage** .. **burn a chip** .. a **program or** which has a special **shock at**	.. you use a higher **voltage** .. **burn a chip** .. a **programmer** which has a special **socket** that

Table 5: *The WERs of CAE without / with context perturbation*

model	**context**	**SWB**	**CH**
baseline RNN-T	-	10.4	16.6
CAE + joint training	hyp	10.1	15.7
	oracle	10.0	15.6
+ context perturbation	hyp	**10.0**	**15.7**
	oracle	10.0	15.7

ing brings in 1.8% relative WER reduction on CH subset with the model hypotheses (denoted as hyp) used as dialogue-context input. Meanwhile, when the context is not available, the joint training alleviates the performance degradation and makes the contextual model more practical.

We evaluate the effectiveness of the context perturbation strategy for the CAE model and present the results in the Table 5. Without loss of generality, during training we conduct the context perturbation together with the joint training strategy. In order to validate the effect of the context mismatch between training and inference, the hyp and oracle dialogue-context are respectively fed to the CAE model. In Table 5, because of the mismatch, on SWB / CH subsets the WERs of the CAE model with hyp context increase absolutely 0.1% compared with models with oracle context. When the context perturbation strategy is adopted, the performance is the same for hyp and oracle context input. And the WER of the CAE model is also reduced.

3.3.3. Comparison of previous E2E ASR systems and our context-aware RNN-T

We summarize all the results of previous E2E ASR systems and our proposed streaming dialogue-context aware RNN-T models on switchboard-2000h task in Table 6 for better comparisons. LM is not used for these models. Our streaming RNN-T baseline achieves WER 10.4% and 16.6% on SWB and CH subset, which is acceptable compared with non-streaming LAS and RNN-T models. As for the context-aware models, the CAE model with 1-turn context achieves 3.8% and 5.4% relative gain relative to the baseline RNN-T. When 3-turn dialogue-context is provided to the CAE, the CAE model achieves 4.8% and 6.0% relative gain. For the sake of simplicity, the 3 turns of dialogue-

Table 6: *The WERs of previous E2E ASR systems and our proposed context-aware RNN-T models*

Model	**Streaming**	**SWB**	**CH**
LAS [33]	N	8.3	15.5
RNN-T [34]		8.5	16.4
RNN-T [35]		6.2	10.9
LAS [15]	N	14.4	21.9
contextual LAS [15]		13.2	21.5
baseline RNN-T	Y	10.4	16.6
CAE 1-turn		10.0	15.7
CAE 3-turn		**9.9**	**15.6**

Table 7: *The WERs of our proposed RNN-T with LM fusion*

Model	LM	**SWB**	**CH**
baseline RNN-T	w/o	10.4	16.6
	NNLM	9.9	15.9
	CLM	**9.9**	**15.6**
CAE 3-turn	w/o	9.9	15.6
	NNLM	9.4	15.6
	CLM	**9.3**	**15.3**

context are concatenated into one sentence. We have tried 5-turn dialogue-context but no more significant gain is found. We assume that more sophisticated context encoding methods are needed, which is our future work. Note that 5-turn context is used in Kim's context LAS model [15].

In Table 4, we provide examples from CH testset. Each column shows reference utterances, the hypothesis of the baseline model, and the hypothesis of our proposed CAE model in chronological order. It is shown that the CAE model is able to benefit from these semantically related words in the historical utterances.

3.3.4. Context-aware LM (CLM) integration

We apply LM shallow fusion during beam search [31]. To take full advantage of the dialogue-context in the conversational ASR system, in addition to the standard NNLM, we also adopt the context-aware LM (CLM) for decoding. The results in Table 7 show that, when shallow fusion with NNLM is applied, the CAE model outperforms the non-contextual model by relative 5.1% / 1.9% on the SWB and CH subsets. We assume this is because the CAE model provides better candidate hypotheses for LM fusion during beam search. Meanwhile, the CLM outperforms NNLM by 1%$\sim$2% for both non-contextual RNN-T and context-aware RNN-T models. Together with the CLM integration, we finally get a 10.6% / 7.8% relative gain in the CAE model over the baseline RNN-T.

4. Conclusions

In this paper, we present a dialogue-context aware streaming RNN-T model for conversational ASR on Switchboard-2000h task. We propose CAP and CAE to combine the dialogue-context and RNN-T in a deep integration manner, which achieves better performance than that of the prior works. With our proposed training strategies including joint training and context perturbation, we further reduce the WERs on the SWB and CH subsets, and eliminate the mismatch of dialogue-context between training and inference. By leveraging dialogue-context, our context-aware RNN-T model yields 4.8% / 6.0% relatively improvement on SWB and CH Hub5'00 test sets with 3-turn dialogue-context. An additional 5.8% / 1.8% relative improvement is achieved when integrating a context-aware LM. Future work includes better utilizing multi-turn dialogue-context and extending our work to other conversation tasks.

5. References

[1] A. Graves, S. Fernández, F. Gomez, and J. 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.

[2] A. Graves, "Sequence transduction with recurrent neural networks," in *Proceedings of the 29th international conference on Machine learning*, 2012.

[3] L. Huang, J. Sun, Y. Tang, J. Hou, J. Chen, J. Zhang, and Z. Ma, "Hmm-free encoder pre-training for streaming RNN transducer," in *Proc. Interspeech*, 2021, pp. 1797–1801.

[4] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2016, pp. 4960–4964.

[5] C.-C. Chiu, T. N. Sainath, Y. Wu, R. Prabhavalkar, P. Nguyen, Z. Chen, A. Kannan, R. J. Weiss, K. Rao, E. Gonina *et al.*, "State-of-the-art speech recognition with sequence-to-sequence models," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4774–4778.

[6] G. Pundak, T. N. Sainath, R. Prabhavalkar, A. Kannan, and D. Zhao, "Deep context: end-to-end contextual speech recognition," in *2018 IEEE spoken language technology workshop (SLT)*. IEEE, 2018, pp. 418–425.

[7] F.-J. Chang, J. Liu, M. Radfar, A. Mouchtaris, M. Omologo, A. Rastrow, and S. Kunzmann, "Context-aware transformer transducer for speech recognition," in *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2021, pp. 503–510.

[8] M. Han, L. Dong, Z. Liang, M. Cai, S. Zhou, Z. Ma, and B. Xu, "Improving end-to-end contextual speech recognition with fine-grained contextual knowledge selection," in *2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 8532–8536.

[9] Z. Chen, M. Jain, Y. Wang, M. L. Seltzer, and C. Fuegen, "End-to-end contextual speech recognition using class language models and a token passing decoder," in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6186–6190.

[10] M. Jain, G. Keren, J. Mahadeokar, G. Zweig, F. Metze, and Y. Saraf, "Contextual RNN-T for open domain ASR," in *Proc. Interspeech*, 2020, pp. 11–15.

[11] Y. He, R. Prabhavalkar, K. Rao, W. Li, A. Bakhtin, and I. McGraw, "Streaming small-footprint keyword spotting using sequence-to-sequence models," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2017, pp. 474–481.

[12] Z. Wu, B. Li, Y. Zhang, P. S. Aleksic, and T. N. Sainath, "Multistate encoding with end-to-end speech rnn transducer network," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7819–7823.

[13] S. N. Ray, M. Wu, A. Raju, P. Ghahremani, R. Bilgi, M. Rao, H. Arsikere, A. Rastrow, A. Stolcke, and J. Droppo, "Listen with intent: Improving speech recognition with audio-to-intent front-end," in *Proc. Interspeech*, 2021, pp. 3455–3459.

[14] S. N. Ray, S. Mitra, R. Bilgi, and S. Garimella, "Improving rnn-t asr performance with date-time and location awareness," in *International Conference on Text, Speech, and Dialogue*. Springer, 2021, pp. 394–404.

[15] S. Kim, "End-to-end speech recognition on conversations," Ph.D. dissertation, Carnegie Mellon University, 2019.

[16] S. Kim and F. Metze, "Dialog-context aware end-to-end speech recognition," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 434–440.

[17] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*. Association for Computational Linguistics, 2019, pp. 4171–4186.

[18] S. Kim, S. Dalmia, and F. Metze, "Gated embeddings in end-to-end speech recognition for conversational-context fusion," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 1131–1141.

[19] A. Kojima, "Large-context automatic speech recognition based on rnn transducer," in *2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2021, pp. 460–464.

[20] W. Xiong, L. Wu, J. Zhang, and A. Stolcke, "Session-level language modeling for conversational speech," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 2764–2768.

[21] D.-R. Liu, C. Liu, F. Zhang, G. Synnaeve, Y. Saraf, and G. Zweig, "Contextualizing asr lattice rescoring with hybrid pointer network language model," in *Proc. Interspeech*, 2020, pp. 3650–3654.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[23] S. Bengio, O. Vinyals, N. Jaitly, and N. Shazeer, "Scheduled sampling for sequence prediction with recurrent neural networks," *Advances in neural information processing systems*, vol. 28, 2015.

[24] J. Godfrey and E. Holliman, "Switchboard-1 release 2 ldc97s62," *Linguistic Data Consortium*, 1993.

[25] G. John and E. Holliman, "Switchboard-1 release 2 ldc97s62," *Web Download. Philadelphia: Linguistic Data Consortium*, p. 35, 1993.

[26] C. Cieri, D. Miller, and K. Walker, "Fisher english training speech parts 1 and 2," *Philadelphia: Linguistic Data Consortium*, 2004.

[27] C. Cieri, D. Graff, O. Kimball, D. Miller, and K. Walker, "Fisher english training speech part 1 transcripts," *Philadelphia: Linguistic Data Consortium*, 2004.

[28] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. E. Y. Soplin, J. Heymann, M. Wiesner, N. Chen *et al.*, "Espnet: End-to-end speech processing toolkit," in *Proc. Interspeech*, 2018, pp. 2207–2211.

[29] I. Turc, M.-W. Chang, K. Lee, and K. Toutanova, "Well-read students learn better: On the importance of pre-training compact models," *arXiv preprint arXiv:1908.08962*, 2019.

[30] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Volume 1*. Association for Computational Linguistics, 2016, pp. 1715–1725.

[31] R. Cabrera, X. Liu, M. Ghodsi, Z. Matteson, E. Weinstein, and A. Kannan, "Language model fusion for streaming end to end speech recognition," *arXiv preprint arXiv:2104.04487*, 2021.

[32] M. Ghodsi, X. Liu, J. Apfel, R. Cabrera, and E. Weinstein, "Rnn-transducer with stateless prediction network," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7049–7053.

[33] C. Weng, J. Cui, G. Wang, J. Wang, C. Yu, D. Su, and D. Yu, "Improving attention based sequence-to-sequence models for end-to-end english conversational speech recognition." in *Proc. Interspeech*, 2018, pp. 761–765.

[34] E. Battenberg, J. Chen, R. Child, A. Coates, Y. G. Y. Li, H. Liu, S. Satheesh, A. Sriram, and Z. Zhu, "Exploring neural transducers for end-to-end speech recognition," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2017, pp. 206–213.

[35] G. Saon, Z. Tüske, and K. Audhkhasi, "Alignment-length synchronous decoding for rnn transducer," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7804–7808.