



# Improving ASR Robustness in Noisy Condition Through VAD Integration

Sashi Novitasari<sup>1,2</sup>, Takashi Fukuda<sup>1</sup>, Gakuto Kurata<sup>1</sup>

<sup>1</sup>IBM Research - Tokyo, Japan

<sup>2</sup>Nara Institute of Science and Technology, Japan

sashi.novitasari.si3@is.naist.jp, fukuda1@jp.ibm.com, gakuto@jp.ibm.com

## Abstract

Automatic speech recognition (ASR) systems are often deployed together with a voice activity detection (VAD) system to run ASR only on the voiced acoustic signals. Although it can maintain ASR performance by removing unnecessary non-speech parts from input audio signals during inference, an error propagates when VAD fails to split speech and non-speech segments correctly. Specifically, because ASR systems are commonly constructed using segmented speech utterances only, many unexpected insertion errors can occur when VAD-segmented utterances contain a long non-speech part or only consist of non-speech. Note VAD is more prone to fail in noisy environments or in unknown acoustic domains, which triggers insertion errors in ASR more prominently. In this paper, we focus on explicitly incorporating VAD information into training of a recurrent neural network transducer (RNN-T) based ASR to make the model more robust to noisy conditions through feature integration and a multi-task learning strategy. A technique is also explored that utilizes audio-only untranscribed data by distilling VAD-related knowledge to the ASR part of the model. By combining the multi-task learning approach with the feature integration architecture, our system yields up to 10% relative improvements in very low signal-to-noise ratio (SNR) conditions compared with the system simply trained on mixed data consisting of speech and long non-speech segments.

**Index Terms:** End-to-end speech recognition, RNN transducer, voice activity detection, noisy speech

## 1. Introduction

Automatic speech recognition (ASR) systems have become prominent in human-machine communication. Recent ASR systems with end-to-end neural network architectures [1, 2, 3] have performed remarkably with less development cost than the conventional hybrid ASR systems. Among well-known neural ASR systems, the recurrent neural network transducer (RNN-T) [4, 5, 6, 7] has been widely used due to it having lower computational cost than systems designed on other architectures, with a competitive performance, and a capability for online speech recognition. ASR is often paired with a voice activity detection (VAD) system [8, 9, 10, 11] that extracts actual speech parts from an input audio signal by removing non-speech parts before the decoding process of ASR starts. Recently, neural network-based VAD has also attracted attention in the speech research community to capture unique properties of speech in various noisy conditions [12, 13, 14].

Although VAD supports the speech recognition process in realistic situations where speakers utter in various times and places, severe recognition errors occur when VAD fails to split the speech and non-speech segments in the input audio. For

This work was done during the first author's internship period in IBM.

example, if the VAD system determines a non-speech audio segment as speech, ASR tries to output a text from an empty speech input. Such errors are triggered because ASR systems are commonly trained with well-segmented speech data. More specifically, training data usually contains short silence regions before and after the actual speech segments, and long silence regions are removed from the training data in advance. Thus, in noisy conditions where VAD performs inaccurately, ASR accuracy can be deteriorated.

Several attempts were previously made to integrate end-to-end ASR and VAD to improve speech segmentation and recognition performance. One conventional method incorporated connectionist temporal classification (CTC)-based ASR with a VAD task, where the speech is segmented by assuming blank labels from the CTC softmax output as speech boundaries [15]. In another approach, a multi-task learning framework for ASR and VAD was also proposed. In this approach, ASR and VAD share the common layers that extract a latent representation from a raw waveform input [16]. This system is optimized by using a combination of ASR and VAD criterion. These previous works mainly focus on ASR in unsegmented long audio in a clean condition. Another work also proposes a multi-task learning between audio-visual ASR and VAD for noisy speech inputs to leverage visual information [17].

In this work, we investigate methods to explicitly leverage VAD information in training RNN-T based ASR to improve the robustness of speech recognition in noisy conditions. While improving the accuracy of VAD has been a common approach to tackle the problem of ASR vulnerability caused by failures of speech segmentation in noisy situations, our method in this paper focuses on improving ASR itself because the ASR robustness to VAD errors also needs to be enhanced. We integrate the VAD information into ASR through feature-level integration and multi-task learning approaches. Multi-task learning in RNN-T has been proposed previously with keyword-spotting [18] and language modeling as a sub-task [19] for rare word recognition. To improve noisy ASR performance, our multi-task learning jointly minimizes ASR RNN-T loss and VAD errors when frame-level speech/non-speech labels generated by convolutional neural network (CNN) based VAD system using spectro-temporal are predicted [12, 20].

In addition, we explore a method to effectively utilize untranscribed audio data in the multi-task learning framework as the auxiliary training data for a VAD component, motivated by knowledge distillation approaches [21, 22, 23]. Normally, preparing transcribed data for ASR training that covers a vast domain of speech is very expensive. The amount of training data for ASR is hence often limited. In contrast, the data collection and labeling for VAD systems is less expensive than for ASR systems. VAD systems trained using the data of diverse environmental domains can perform well in various acoustic conditions. We hypothesize that the VAD models contain a lot

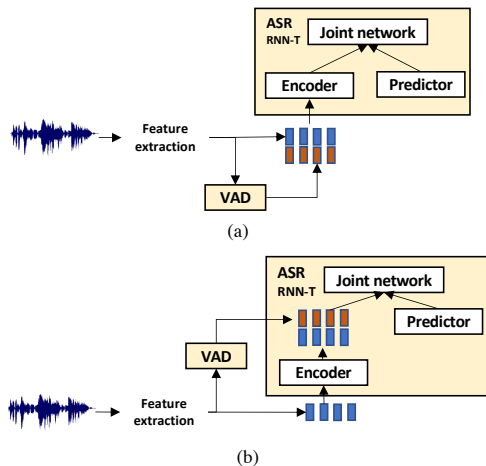


Figure 1: Single-task ASR with VAD feature integration through concatenation at (a) pre-encoder and (b) post-encoder.

of information representing surrounding acoustic environments to distinguish speech from non-speech, which can also be useful for improving the robustness of the ASR model. In this paper, we propose a method to distill abundant acoustic environmental knowledge contained in well-trained VAD models into ASR networks without using any additional transcribed data for improving ASR robustness. In the experiments with English telephone conversations with long silence portions, our experimental result shows that, by using the auxiliary VAD training data, our system improved the ASR noise robustness also in new acoustic domains.

## 2. Proposed Framework

RNN-T model consists of an encoder network, prediction network, and joint network. Given a speech feature input sequence  $\mathbf{x} = (x_1, \dots, x_T)$  of length  $T$ , RNN-T outputs the text token sequence  $\mathbf{y} = (y_1, \dots, y_U)$  of length  $U$  by modeling the conditional distribution  $p(\mathbf{y}|\mathbf{x})$  of input and output. We present two approaches, feature integration and multi-task training, in transferring VAD-based knowledge to the RNN-T ASR system.

### 2.1. Single-task ASR with VAD Feature Integration

We perform a feature-level integration by concatenating voice activity class probabilities generated from separately trained VAD with ASR features. The VAD system predicts the sequence of voice activity class  $\mathbf{v} = (v_1, \dots, v_T)$  of length  $T$  from a speech frame sequence  $\mathbf{x}$  with the same length. In our framework, features are concatenated between the VAD output probability  $p(\mathbf{v}|\mathbf{x})$  and the ASR feature of the corresponding speech frame. In this work, two integration approaches based on positions in the network are investigated:

- a) **Pre-encoder:** VAD output probability  $p(\mathbf{v}|\mathbf{x})$  is concatenated with the input feature for RNN-T  $\mathbf{x}$  before the encoder, shown in Fig. 1(a), formally written as

$$\mathbf{x}_c = ((x_1, p(v_1|\mathbf{x})), \dots, (x_T, p(v_T|\mathbf{x}))), \quad (1)$$

where  $\mathbf{x}_c$  is the encoder input feature after the concatenation.

- b) **Post-encoder:** Here the feature-level integration is made between the VAD output probability  $p(\mathbf{v}|\mathbf{x})$  and the RNN-T encoder output  $\mathbf{h} = (h_1, \dots, h_T)$ , as shown in Fig. 1(b), in the form

$$\mathbf{h} = \text{Encoder}(\mathbf{x}), \quad (2)$$

$$\mathbf{h}_c = ((h_1, p(v_1|\mathbf{x})), \dots, (h_T, p(v_T|\mathbf{x}))), \quad (3)$$

where  $\mathbf{h}_c$  is the integrated features that will be passed to the joint network during the decoding.

## 2.2. Multi-task ASR-VAD

### 2.2.1. Architecture

In our proposed method, encoder layers of RNN-T for ASR processing are shared with those for VAD processing as the sub-task. We investigate four multi-task architectures on the basis of how many encoder layers in the RNN-T are shared.

- a) **MTL 1 - pre-encoder sharing:** An additional network is appended to RNN-T before the encoder module (Fig. 2(a)), from which the shared input representation for ASR and VAD will be produced. The shared network consists of a stack of fully connected neural network (FC) layers with the hyperbolic tangent function. The VAD component as a sub-task in this framework consists of a CNN.
- b) **MTL 2 - partial encoder sharing:** ASR and VAD share a part of the RNN-T encoder from the bottom layer, as shown in Fig 2(b). The VAD branch is followed by a stack of FC layers that predict VAD classes of the input.
- c) **MTL 3 - full encoder sharing:** ASR and VAD use all the encoder layers in RNN-T as the shared network, which is shown in Fig. 2(c). The VAD component is the same as that in MTL 2.
- d) **MTL 4 - full encoder sharing with feature integration:** We train the entire network with the multi-task learning approach by using all the encoder layers as the shared layers and with the feature integration, shown in 2(d). The VAD soft outputs are projected into a vector with the same dimension as the RNN-T encoder output using an FC layer. These two sequences are merged by using an element-wise summation operation.

ASR-VAD multi-task learning here is done using a shared transcribed data between ASR and VAD tasks. The network is optimized sequentially on the basis of the components starting from ASR only, VAD only, and ending with ASR-VAD joint optimization.

### 2.2.2. Multi-task training with auxiliary VAD data

In addition to the multi-task learning with shared transcribed data, we also propose utilizing auxiliary untranscribed audio-only data for an optimization of the network related to the VAD task to improve ASR performance. ASR-VAD multi-task learning with the auxiliary VAD data is performed through three training steps:

1. RNN-T optimization for the ASR task only by freezing the VAD parameters, using the transcribed data.
2. VAD optimization for the network pre-trained in the step 1 by freezing ASR parameters and the shared layers, using both transcribed and auxiliary untranscribed data.
3. ASR-VAD joint optimization. For each training epoch:
  - 1) Update the VAD and the shared layer parameters by using the untranscribed VAD training data with a weighted loss

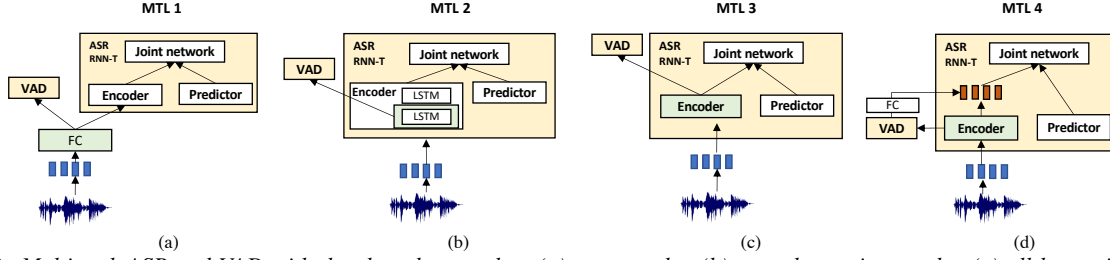


Figure 2: Multi-task ASR and VAD with the shared network at (a) pre-encoder; (b) some layers in encoder; (c) all layers in encoder without feature integration, and (d) all layers in encoder with feature integration.

$$Loss = \beta \cdot Loss_{VAD}. \quad (4)$$

where  $\beta$  is a hyper-parameter for VAD task loss weight for the joint optimization in the next step.

- Update all ASR and VAD parameters using the shared transcribed data and the joint weighted loss

$$Loss = \alpha \cdot Loss_{ASR} + \beta \cdot Loss_{VAD}, \quad (5)$$

where  $\alpha$  is a hyper-parameter for ASR task loss weight.

Unlike the training steps in Section 2.2.1, step 3-1) with Eq.(4) here is especially unique to the better use of untranscribed data, which is crucial for enhancing the robustness of the speech recognition side.

### 2.2.3. VAD training through label distillation

In the VAD sub-task optimization used in the above proposed multi-task training, we use pseudo labels generated from a separately well-trained VAD model. From a different view point, this is regarded as knowledge distillation from VAD to ASR. Knowledge distillation is a technique to mimic complicated teacher networks with a simple student network. The separate VAD model, a teacher VAD model, is trained with a large amount of training data to generate better pseudo soft labels for a student VAD where the student here is a sub-task VAD network connected to the RNN-T ASR network. Thus, distillation is performed through the pseudo VAD labels  $\tilde{\mathbf{v}}$  obtained from the acoustic features  $\mathbf{x}$  in the training material, expressed as

$$\mathbf{v} = \arg \max_{\tilde{\mathbf{v}}} p(\tilde{\mathbf{v}}|\mathbf{x}), \quad (6)$$

where  $p(\tilde{\mathbf{v}}|\mathbf{x})$  is a VAD class posterior predicted by the teacher VAD. The  $\mathbf{v}$  is then utilized as the target label of the student VAD in the multi-task framework given  $\mathbf{x}$ .

## 3. Experiments

In our first set of experiments, we used the Switchboard (SWB) corpus consisting of 300-hour multi-speaker American-English speech from telephone conversations. We augmented this data by including the non-speech segments longer than 2 sec in the training material, in which these are usually discarded in the common training setting. After the data augmentation, we have 88 hours of non-speech only data and 599 hours of utterances mixed with speech and non-speech segments, in addition to the standard 267 hours of the SWB data. For the second set of experiments, we prepared the auxiliary VAD training data (VAD aux-data) for the updates with the VAD criterion in the multi-task learning framework as described in Section 2.2.2, which consists of 200 hours of English call center (CC) conversations together with 200 hours of non-speech. No transcription is available for this training set. Environmental noises were

Table 1: ASR (WER%) and VAD (EER%) performance comparison on SWB and CH under noisy conditions at an average SNR 4.2 dB. MTL  $\alpha = 1.0$ . Pre-trained teacher VAD EER 24.6 %.

System	ASR						VAD
	Speech + Non-speech			Speech-only			
	SWB	CH	Avg.	SWB	CH	Avg.	
<b>Standard RNN-T</b>							
No aug	20.5	34.4	27.5	13.5	29.7	21.6	-
Aug	15.5	29.3	22.4	13.9	33.9	23.9	-
<b>Single-task ASR-VAD feature integration</b>							
Pre-enc	15.9	32.6	24.3	14.5	45.2	29.9	-
Post-enc	15.1	29.4	22.2	14.1	34.1	24.1	-
<b>Multi-task ASR-VAD (<math>\beta=0.05</math>)</b>							
MTL 1	15.1	29.5	22.3	13.9	34.1	24.0	23.9
MTL 2	15.1	30.6	22.9	13.8	34.9	24.4	22.6
MTL 3	15.1	30.0	22.5	13.8	37.1	25.5	22.0
MTL 4	15.1	29.3	22.2	13.9	34.3	24.1	21.5
<b>Multi-task ASR-VAD (<math>\beta=0.01</math>)</b>							
MTL 3	15.0	29.2	22.1	13.9	34.6	24.3	22.3
MTL 4	14.8	29.2	22.0	13.8	34.6	24.2	21.1
<b>Multi-task ASR-VAD + VAD aux-data (<math>\beta=0.01</math>)</b>							
MTL 1	15.1	30.6	22.9	13.8	36.2	25.0	29.1
MTL 2	14.9	28.6	21.8	13.8	31.9	22.9	22.4
MTL 3	15.6	27.7	21.7	13.9	31.8	22.9	22.2
MTL 4	15.0	28.2	21.6	13.7	31.9	22.8	22.3

Table 2: WER(%) comparison on non-speech (noisy speech only) input. MTL  $\alpha = 1.0$  and  $\beta=0.01$ .

System	SWB	CH	Avg.
Standard RNN-T (No aug)	123.3	182.3	152.8
Standard RNN-T (Aug)	6.1	20.8	13.5
Post-enc	3.0	13.1	8.1
MTL 4	7.3	18.1	12.7
MTL 4 + VAD aux-data	4.7	20.6	12.7

added to these training sets for the model training. The average speech-to-noise ratio (SNR) was 14 dB.

Our RNN-T implementation followed the same configuration presented in the previous work [7], whose model is composed of 6 bidirectional long short-term memory (Bi-LSTM) encoder layers with 640 cells per layer per direction and a single unidirectional LSTM prediction layer with only 1024 cells. The joint network projects the 1280-dimensional stacked encoder vectors from the last layer and the 1024-dimensional prediction net embedding to 256 dimensions and combines the projected vectors. After the application of a hyperbolic tangent, the output is projected to 42 logits followed by a softmax layer corresponding to 41 characters plus BLANK. We extract 40-dimensional speaker independent log-Mel filterbank features every 10 ms as ASR features. After utterance level mean and global variance normalization, these features are augmented with delta and double delta coefficients. The independent CNN-based VAD to generate frame-level VAD labels, which were utilized in the

Table 3: WER(%) comparison on audio input segmented using independent CNN-based VAD at an average SNR 4.2 dB. MTL  $\alpha = 1.0$  and  $\beta=0.01$ .

System	Speech + Non-speech		
	SWB	CH	Avg.
Standard RNN-T (No aug)	21.8	35.9	28.9
Standard RNN-T (Aug)	19.9	35.3	27.6
MTL 4 + VAD aux-data	18.7	31.9	25.3

Table 4: WER(%) investigation of acoustic customization capability in different SNR (dB) settings. MTL  $\alpha = 1.0$  and  $\beta=0.01$ .

System	SNR	Speech + Non-speech		Speech-only	
		SWB+ CH	CC	SWB+ CH	CC
Standard ASR (Aug)	14.4	15.8	66.1	15.2	54.9
Post-enc		15.8	64.8	15.1	54.8
MTL 4 + VAD aux-data		16.0	64.0	14.7	53.7
Standard ASR (Aug)	4.2	22.4	73.3	23.9	65.6
Post-enc		22.2	71.5	24.1	64.0
MTL 4 + VAD aux-data		21.6	70.8	22.8	62.9
Standard ASR (Aug)	1.4	35.3	79.5	37.5	74.0
Post-enc		34.9	78.6	38.1	72.7
MTL 4 + VAD aux-data		33.9	77.9	36.4	71.4

feature integration and multi-task learning systems, was constructed with settings similar to those in [12]. It consists of 4 convolutional layers with the input channels 3, 16, 32, and 2 respectively from the first to last layer and the output class dimension of 3 in classifying the speech frame into speech, non-speech, and music classes. This isolated VAD was trained on more than 2000-hour English speech data consisting of various ASR domains including both spontaneous and read speech. On the other hand, the VAD component is a sub-task for MTL 2, MTL 3, and MTL 4 composed of three stacks of FC layers. In MTL 2, the first three encoder layers were shared between ASR and VAD tasks. The learning rate for ASR was  $2e-4$ , and that for VAD in the multi-task learning framework was  $2e-5$ . Both systems were optimized by using the stochastic gradient descent (SGD) with a batch size of 64.

### 3.1. Baseline

Experiments were carried out using speech-only segments and also those combined with the non-speech segments simulating VAD prediction errors. Results of these experiments are shown in Tables 1 and 2 with ASR word error rate (WER %) and frame-level VAD equal error rate (EER%). The baseline and proposed systems were evaluated on modified SWB and Callhome (CH) test sets created by artificially adding various lengths of non-speech segments before speech, after speech, or between two speech segments. Non-speech segments added to each test utterance are 5.5 sec on average. Realistic environmental noises were also added to these test sets. In addition, speech-only and noise-only (non-speech) tests were also conducted.

Our experimental results in Tables 1 and 2 show that the standard RNN-T tagged as “No aug” trained with the original SWB training corpus performed well under the manually-segmented ideal speech-only input condition, but performance degraded drastically in more realistic cases with the speech combined with long non-speech portions. Under this condition, the text decoded by “Standard RNN-T (No aug)” contains many unexpected insertion errors on the non-speech parts. In contrast, adding non-speech segments to the training dataset (“Standard

RNN-T (Aug)”) significantly decreased the number of those errors, which can be also seen in Table 2.

### 3.2. Feature Integration and Multi-task Learning

In all test conditions, the post-encoder feature integration performs better than the pre-encoder feature integration (Table 1). Although WERs by the post-encoder integration on “Speech + Non-speech” and “Speech-only” input cases were similar to “Standard RNN-T (Aug)”, this technique significantly reduced insertion errors in the test case of non-speech only as shown in Table 2.

Next we consider methods based on multi-task learning. Experimental results are also tabulated in Tables 1 and 2. When the VAD auxiliary data was not utilized, MTL 4 ( $\beta=0.01$ ) provided the largest improvement in the “Speech + Non-speech” test case. Further improvements were obtained by utilizing auxiliary VAD training data (VAD aux-data) in MTL 2, MTL 3, and MTL 4. They improved WERs for not only “Speech + Non-speech” but also “Speech-only” test cases compared with models without auxiliary VAD data. “MTL 4 + VAD aux-data” showed the best performance yielded relative improvements of 21.5 % and 3.6 % in the “Speech + Non-speech” test case compared with the standard RNN-T without and with data augmentation, respectively. Also, “MTL 4 + VAD aux-data” provided a relative improvement of 4.6% in the speech-only test case compared with “Standard RNN-T (Aug)”.

We further conducted experiments with audio signals automatically segmented by the separate VAD system, which was used for generating VAD labels for the proposed method. Results are shown in Table 3. Here we focus on “Speech + Non-speech” test cases. Because various kinds of segmentation errors in the VAD results at low SNRs included the classification of speech segment as noise, the absolute WERs here were larger than those in Table 1 on average. However, the proposed method showed consistent gains over the baseline systems.

### 3.3. Acoustic Customization Through VAD Optimization

We investigated the capability of acoustic customization via the VAD sub-task optimization with additional test data (CC) that is the same domain as the VAD auxiliary data. WERs on different SNRs were also investigated. Results are shown in Table 4. CC is an acoustically out-of-domain test set, and hence absolute WERs are high. By adding VAD auxiliary data to the training data, “MTL 4 + VAD aux-data” yielded improvements for both CC and SWB/CH test sets in every SNR setting. Interestingly, the gap in WER between “standard RNN-T (Aug)” and “MTL 4 + VAD aux-data” tends to become larger as the SNR decreases. In a situation with acoustically challenging data to which the unsupervised and semi-supervised training of ASR [24, 25, 26, 27] cannot be applied, our proposed method can show a promising improvement, which is relatively computationally inexpensive. These experimental results showed that the proposed model was able to enhance the robustness against noisy environments.

## 4. Conclusions

We presented several methods to explicitly leverage VAD information for training RNN-T based ASR through feature integration and multi-task learning for noise robustness. With the combination of distilling VAD knowledge to ASR and exploiting additional VAD-specific data while training an ASR model, our multi-task system combined with ASR-VAD feature integration outperforms other systems in noisy conditions and new acoustic domains.

## 5. References

- [1] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proceedings of the 31st International Conference on Machine Learning*, 2014, pp. 1764–1772. [Online]. Available: <https://proceedings.mlr.press/v32/graves14.html>
- [2] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *Proc. ICASSP 2016*, 2016, pp. 4960–4964.
- [3] L. Dong, S. Xu, and B. Xu, "Speech-transformer: A no-recurrence sequence-to-sequence model for speech recognition," in *Proc. ICASSP 2018*, 2018, pp. 5884–5888.
- [4] A. Graves, "Sequence transduction with recurrent neural networks," *arXiv preprint arXiv:1211.3711*, 2012.
- [5] J. Li, R. Zhao, H. Hu, and Y. Gong, "Improving rnn transducer modeling for end-to-end speech recognition," in *Proc. ASRU 2019*. IEEE, 2019, pp. 114–121.
- [6] B. Li, S.-y. Chang, T. N. Sainath, R. Pang, Y. He, T. Strohmaier, and Y. Wu, "Towards fast and accurate streaming end-to-end ASR," in *Proc. ICASSP 2020*, 2020, pp. 6069–6073.
- [7] G. Saon, Z. Tüske, D. Bolanos, and B. Kingsbury, "Advancing rnn transducer technology for speech recognition," in *Proc. ICASSP 2021*. IEEE, 2021, pp. 5654–5658.
- [8] ITU-T, "A silence compression scheme for G.729 optimized for terminals conforming to recommendation v.70," *ITU-T Recommendation G.729 Annex B*, 1996. [Online]. Available: <https://ci.nii.ac.jp/naid/10027284458/en/>
- [9] K. H. Woo, T. Y. Yang, K.-J. Park, and C. Lee, "Robust voice activity detection algorithm for estimating noise spectrum," *Electronics Letters*, vol. 36, pp. 180–181, Feb 2000.
- [10] B. Kingsbury, G. Saon, L. Mangu, M. Padmanabhan, and R. Sarikaya, "Robust speech recognition in noisy environments: The 2001 ibm spine evaluation system," in *Proc. ICASSP 2002*, vol. 1, 2002, pp. I-53–I-56.
- [11] J. Dines, J. Vepa, and T. Hain, "The segmentation of multi-channel meeting recordings for automatic speech recognition," IDIAP, Tech. Rep., 2006.
- [12] S. Thomas, S. Ganapathy, G. Saon, and H. Soltau, "Analyzing convolutional neural networks for speech activity detection in mismatched acoustic conditions," in *Proc. ICASSP 2014*, 2014, pp. 2519–2523.
- [13] J. Kim and M. Hahn, "Voice activity detection using an adaptive context attention model," *IEEE Signal Processing Letters*, vol. 25, no. 8, pp. 1181–1185, 2018.
- [14] S. Braun and I. Tashev, "On training targets for noise-robust voice activity detection," in *2021 29th European Signal Processing Conference*, 2021, pp. 421–425.
- [15] T. Yoshimura, T. Hayashi, K. Takeda, and S. Watanabe, "End-to-end automatic speech recognition integrated with CTC-based voice activity detection," in *Proc. ICASSP 2020*, 2020, pp. 6999–7003.
- [16] M. Li, S. Zhou, and B. Xu, "Long-running speech recognizer: An end-to-end multi-task learning framework for online ASR and VAD," *arXiv preprint arXiv:2103.01661*, 2021.
- [17] F. Tao and C. Busso, "End-to-end audiovisual speech recognition system with multitask learning," *IEEE Transactions on Multimedia*, vol. 23, pp. 1–11, 2021.
- [18] Y. Tian, H. Yao, M. Cai, Y. Liu, and Z. Ma, "Improving rnn transducer modeling for small-footprint keyword spotting," in *Proc. ICASSP 2021*, 2021, pp. 5624–5628.
- [19] C.-H. H. Yang, L. Liu, A. Gandhe, Y. Gu, A. Raju, D. Filimonov, and I. Bulyko, "Multi-task language modeling for improving speech recognition of rare words," in *Proc. ASRU 2021*. IEEE, 2021, pp. 1087–1093.
- [20] A. Sehgal and N. Kehtarnavaz, "A convolutional neural network smartphone app for real-time voice activity detection," *IEEE Access*, vol. 6, pp. 9017–9026, 2018.
- [21] G. Hinton, O. Vinyals, J. Dean *et al.*, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, vol. 2, no. 7, 2015.
- [22] H. Futami, H. Inaguma, S. Ueno, M. Mimura, S. Sakai, and T. Kawahara, "Distilling the Knowledge of BERT for Sequence-to-Sequence ASR," in *Proc. Interspeech 2020*, 2020, pp. 3635–3639.
- [23] X. Xu, H. Dinkel, M. Wu, and K. Yu, "A Lightweight Framework for Online Voice Activity Detection in the Wild," in *Proc. Interspeech 2021*, 2021, pp. 371–375.
- [24] D.-R. Liu, C.-Y. Yang, S.-L. Wu, and H.-Y. Lee, "Improving unsupervised style transfer in end-to-end speech synthesis with end-to-end speech recognition," in *2018 IEEE Spoken Language Technology Workshop (SLT)*, 2018, pp. 640–647.
- [25] A. Baevski, W.-N. Hsu, A. Conneau, and M. Auli, "Unsupervised speech recognition," *Advances in Neural Information Processing Systems*, vol. 34, 2021.
- [26] J. Drexler and J. Glass, "Combining end-to-end and adversarial training for low-resource speech recognition," in *2018 IEEE Spoken Language Technology Workshop (SLT)*, 2018, pp. 361–368.
- [27] A. Tjandra, S. Sakti, and S. Nakamura, "Machine speech chain," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 976–989, 2020.