INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Target Vocabulary Recognition Based on Multi-Task Learning with Decomposed Teacher Sequences

Aoi Ito[1,2], *Tatsuya Komatsu*[1], *Yusuke Fujita*[1], *Yusuke Kida*[1]

[1]LINE Corporation, Japan
[2]Hosei University, Japan

Abstract

This paper proposes a method for target vocabulary recognition based on multi-task learning with decomposed teacher sequences. The proposed method first decomposes teacher sequences into the target vocabulary and the non-target vocabulary sequences. Then, multi-task learning is performed by calculating losses for both the target vocabulary sequence and the non-target vocabulary sequence. By utilizing information from both target and non-target vocabulary, our proposed method provides more stable training and more accurate recognition of target vocabulary than single-task learning using only the target vocabulary. Experiments conducted on the Corpus of Spontaneous Japanese (CSJ) dataset, using numerals and katakana as target vocabulary, demonstrate the effectiveness of our proposed method. The results show a maximum CER improvement rate of 27% for katakana and 34% for numerals in target vocabulary recognition, as well as an 84% reduction in insertion errors in non-target vocabulary utterances.

Index Terms: speech recognition, connectionist temporal classification, multi-task learning, target vocabulary recognition

1. Introduction

End-to-end speech recognition has evolved with the development of neural networks, and research has been conducted in various directions [1, 2, 3, 4, 5], including Connectionist temporal classification (CTC) [6, 7, 8, 9, 10], RNN-Transducer [11, 12, 13], and attention-based encoder-decoder architectures [14, 15, 16, 17]. Generally, speech recognition aims to transcribe all the content included in the input utterance accurately. However, not all applications require transcription of the entire speech. In some cases, only specific vocabulary needs to be accurately recognized, while the rest of the speech can be ignored.

In industrial applications, there are cases where only specific vocabulary contained in the speech needs to be extracted. For instance, extracting phone numbers or passwords during automated telephone answering, or extracting nouns related to restaurant menus in restaurant reservations. In such applications, the accuracy of the target vocabulary recognition is crucial, and non-target vocabulary can lead to errors such as insertion, hindering the task. Therefore, the development of techniques that can accurately recognize target vocabulary only can greatly enhance the efficiency and effectiveness of these applications.

For improving target vocabulary recognition, keyword spotting [18, 19, 20] can be associated with the task. Many approaches to keyword spotting involve fine-tuning pre-trained ASR models to better detect specific keywords. These approaches are generally focused on detecting specific predefined keywords, such as wake-up words, and are not well-suited for detecting any arbitrary keyword that may be included in the target vocabulary. Some techniques have been proposed that can detect arbitrary keywords using the query-by-example framework [21, 22]. These techniques allow for the identification of any keyword, rather than just predefined ones, by using a sample of the keyword as a reference. However, the model requires a dataset containing speech with specific vocabulary only, making it difficult to train on natural speech datasets. Moreover, the model needs to estimate the location of keywords in the utterance during inference and may not handle multiple keywords within a single utterance.

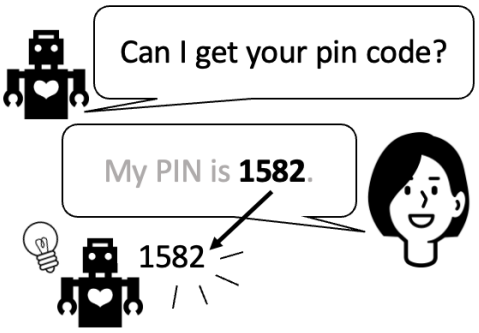

Figure 1: *Automatic telephone assistance that asks for the user's PIN. In this case, content of speech is unnecessary if the PIN information of the user can be obtained.*

Another related field is end-to-end spoken language understanding (SLU), a task of identifying speaker intent, specific named entities, and grammatical information from speech. Researchers in this field have explored various approaches, including the use of Transformer without using automatic speech recognition (ASR) [23], transfer learning of ASR pretraining parameters to SLU models [24], and the use of self-supervised pretraining models as feature extractors [25]. These techniques can produce rich outputs, extracting information beyond that of speech recognition only. However, there is little discussion about the recognition accuracy of the output itself. Additionally, the process of model training can be complex, often involving the combination or fine-tuning of pre-trained models with task-specific datasets. Omachi et al. [26] proposed a method that adds grammatical-information tags to transcribed labels and trains an end-to-end ASR model to output both transcription and tag information simultaneously. However, this method requires newly created transcribed labels with added tag information for training the end-to-end model, which increases the difficulty of the task. While this approach is successful in producing rich output information, the speech recognition accuracy

10.21437/Interspeech.2023-2257

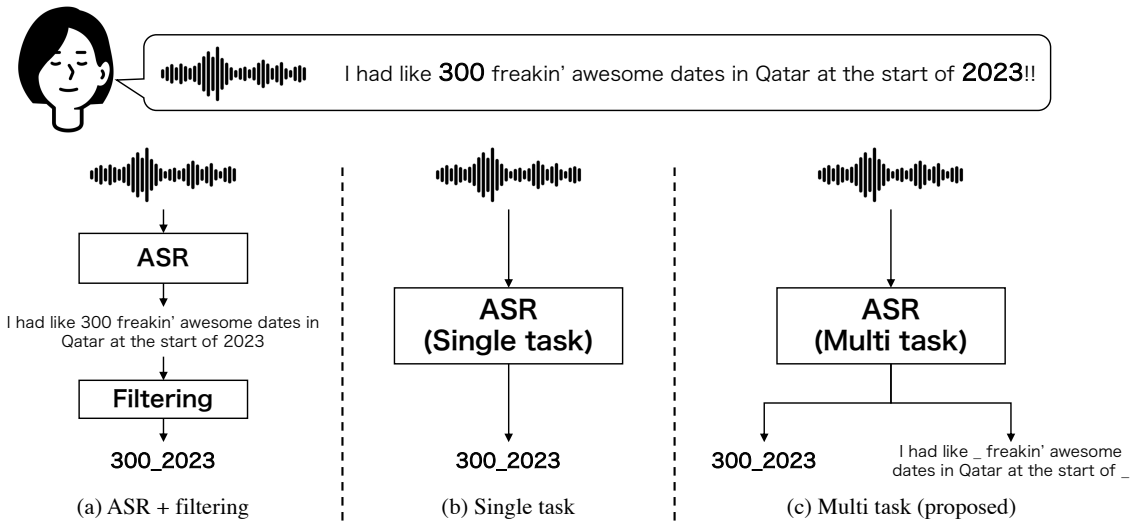

Figure 2: *Example of target vocabulary (numerals) extraction; (a) applying target vocabulary filtering as post-processing to speech recognition output, (b) a single-task trained ASR model using only the target vocabulary as teacher data, (c) the proposed multitask network in which not only the target vocabulary but also the non-target vocabulary is output in parallel. Note that the token "_" represents the non-target vocabulary sequence.*

is decreased as a result.

We propose a new approach for target vocabulary extraction that is both efficient and simple. First, this paper sets up the problem of target vocabulary recognition and discusses the underlying approach and its challenges. Then, we describe a novel multitasking network for target vocabulary recognition that also uses information from non-target vocabulary, which has not been considered previously. By using both target and non-target vocabulary information, the proposed multitasking network is more stable and easy to learn. Experiments conducted on the Corpus of Spontaneous Japanese (CSJ) dataset, using numerals and katakana as target vocabulary, demonstrate the effectiveness of our proposed method. The results show a maximum CER improvement rate of 27% for katakana and 34% for numerals in target vocabulary recognition, as well as an 84% reduction in insertion errors in non-target vocabulary utterances.

2. Problem setting

Speech recognition is a technology that maps input audio $X \in \mathbb{R}^{T \times D}$ to a word sequence $Y \in \mathcal{V}^L$ corresponding to its spoken content, given a vocabulary set $\mathcal{V}$ and the sequence length L. T denotes the number of temporal frames and D denotes the feature dimension. In this paper, we consider the case where we want to output only the sequence composed of a specific sub-vocabulary set, i.e., the target-vocabulary-word sequence $Y_{\mathrm{tgt}} \in \mathcal{V}_{\mathrm{tgt}} \subset \mathcal{V}$. Fig. 2 illustrates an example where the numerals are the target vocabulary.

Fig 2-(a) is the most straightforward approach which is to apply standard speech recognition with the full vocabulary and filter the output sequence $\hat{Y}$ using post-processing techniques to obtain a sequence $\hat{Y}_{\mathrm{tgt}}$ that consists only of the target vocabulary. However, in this approach, errors in speech recognition propagate to the subsequent filtering stages, leading to accuracy degradation.

Fig 2-(b) is an end-to-end approach that directly outputs the sequence $\hat{Y}_{\mathrm{tgt}}$. General keyword spotting is also considered to be included in this approach. This ASR model can be obtained in single-task training with a new teacher sequence Y_{tgt} consisting only of the target vocabulary by filtering the original teacher sequence Y with full vocabulary. One of the limitations of the approach is that the context of the speech cannot be taken into account because texts belonging to the non-target vocabulary are discarded. This can lead to recognition errors, especially insertion errors in utterances that do not contain target vocabulary, as the models may struggle to differentiate between similar-sounding words without context knowledge. Furthermore, this approach has problems in the training stage. Because the models have to learn to differentiate between the target vocabulary and a large number of possible non-target vocabulary only with Y_{tgt}, which makes it difficult to train the model from scratch, as the conventional methods often rely on fine-tuning with the ASR model pre-trained with full vocabulary Y.

3. Proposed Method

In this section, we describe the proposed teacher sequence decomposition and the multi-task model shown in Fig 2-(c). The proposed method first decomposes the teacher sequence Y into two sequences; a target vocabulary sequence Y_{tgt} and a non-target vocabulary sequence $Y_{-\mathrm{tgt}}$. Then, the ASR model is trained as a multi-task network for the two sequences.

3.1. Teacher sequence decomposition

The proposed method decomposes the teacher sequence Y into the target vocabulary sequence Y_{tgt} and the non-target vocabulary sequence $Y_{-\mathrm{tgt}}$. Elements not included in the corresponding vocabulary are replaced by <unk> tokens representing unknown vocabulary.

Here is an example of a teacher sequence for Japanese when numerals are set as the target vocabulary.

Full Vocab. : 私の暗証番号は1582です
(My PIN is 1582)
Target : <unk> 1582 <unk>
Non-target : 私の暗証番号は <unk> です

The next example is a teacher sequence when katakana is set as the target vocabulary.

Full Vocab. : カタールではデーツを楽しみました[a]
(I really enjoyed dates in Qatar)
Target : カタール <unk> デーツ <unk>
Non-target : <unk> では <unk> を楽しみました

[a]The underlined characters are in katakana. カタール represents Qatar and デーツ represents dates.

Katakana is a type of Japanese script that is used primarily for writing loanwords, foreign names and borrowed words, and onomatopoeic expressions [27, 28, 29]. It is one of the three scripts used in written Japanese, alongside hiragana and kanji. Katakana consists of 46 characters, each representing a syllable, and is visually characterized by its angular and straight-lined shapes. The use of katakana can also indicate emphases, such as when emphasizing certain words in a sentence or advertising.

3.2. Network architecture and loss functions

The network architecture of the proposed method is shown in Fig 3. In this study, we use a multi-task network in which a single encoder has multiple CTC decoders for both the target sequence and the non-target sequence in parallel. We adopt the Conformer encoder [12] for acoustic encoding and the CTC decoder [30] for decoding.

Let us consider an N-layer Conformer encoder. Let $X^{(0)} = X$ be the input acoustic feature sequence, and let $Z^{(n)}$ be the output of the n-th layer of the Conformer encoder. The input and output of each layer are represented as follows:

$$X^{(n)} = \mathrm{Encoder}^{(\mathrm{n})}(\mathrm{X}^{(\mathrm{n}-1)})(1 \leq \mathrm{n} \leq \mathrm{N}). \quad (1)$$

The final layer output of the Conformer encoder $Z^{(N)}$ is input to two types of CTC decoders, which correspond to the sequence of the target vocabulary and the sequence of the non-target vocabulary, respectively.

$$\tilde{Y}^{(c)}_{\mathrm{tgt}} = \mathrm{Softmax}(\mathrm{Linear}_{\mathrm{tgt}}(X^{(N)})) \quad (2)$$

$$\tilde{Y}^{(c)}_{-\mathrm{tgt}} = \mathrm{Softmax}(\mathrm{Linear}_{-\mathrm{tgt}}(X^{(N)})) \quad (3)$$

Here, $\mathrm{Linear}_{\mathrm{tgt}/-\mathrm{tgt}}(\cdot)$ is a linear layer that projects an input vector to the target/non-target vocabulary space, and $\hat{Y}^{(c)}_{\mathrm{tgt}}$ and $\hat{Y}^{(c)}_{-\mathrm{tgt}}$ are estimates of the character sequences of the target and non-target vocabulary set, respectively. The two types of sequences obtained are used to calculate the CTC loss with their corresponding teacher sequences.

$$\mathcal{L}_{\mathrm{ctc}} = \lambda \mathcal{L}_{\mathrm{ctc}}(Y^{(c)}_{\mathrm{tgt}}, \tilde{Y}^{(c)}_{\mathrm{tgt}}) + (1-\lambda)\mathcal{L}_{\mathrm{ctc}}(Y^{(c)}_{-\mathrm{tgt}}, \tilde{Y}^{(c)}_{-\mathrm{tgt}}). \quad (4)$$

Here, $Y^{(c)}_{-\mathrm{tgt}}$ is a character sequence corresponding to the word sequence $Y_{-\mathrm{tgt}}$ and λ is a weight parameter for the CTC loss of the target vocabulary set and its complement set, respectively.

In conventional single-task learning, only $Y^{(c)}_{\mathrm{tgt}}$ was used for the loss calculation, and the model could not learn from the information of $Y^{(c)}_{-\mathrm{tgt}}$, which is outside of the target vocabulary. This can lead to a vulnerability to insertion errors for non-target vocabulary sequences during inference, as well as an inability to capture the difference between target and non-target vocabulary sequences during training, potentially resulting in training failure.

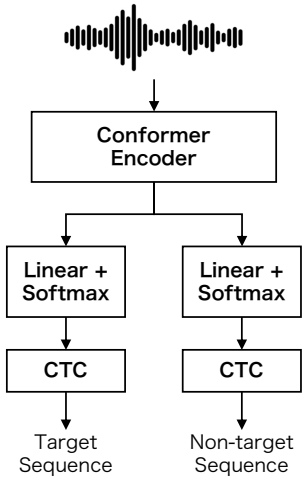

Figure 3: *The proposed multi-task network in which a single encoder has multiple CTC decoders for both the target sequence and the non-target sequence in parallel.*

4. Experiments

We conducted experiments with two different target vocabularies, *katakana*, *numerals*, to evaluate the accuracy of target vocabulary extraction. We used the Corpus of Spontaneous Japanese (CSJ) dataset and decomposed the original teacher sequences with full vocabulary into the target vocabulary and non-target vocabulary based on morphological analysis.

4.1. Dataset and teacher sequences

The Corpus of Spontaneous Japanese (CSJ) [31] dataset is used for the experiments. The training data consisted of 403,071 utterances, and the validation data consisted of 4,000 utterances. We used the official evaluation dataset provided by CSJ for evaluation, which consisted of three evaluation sets: Eval1 (1,272 utterances), Eval2 (1,292 utterances), and Eval3 (1,385 utterances), with a total of 3,949 utterances. Eval1 and Eval2 are recordings of academic presentations, while Eval3 consists of simulated lectures.

We created new teacher sequences by extracting target vocabularies from the original teacher sequences in CSJ. First, we performed morphological analysis on all utterances in CSJ using MeCab [32]. MeCab is an open-source morphological analysis engine that uses Conditional Random Fields (CRF) for parameter estimation. Next, we extracted only the words that MeCab identified as nouns, and then further removed only those that matched the Unicode of Katakana/numeral using regular expressions. At this time, the subset that did not match such as parts of characters other than target vocabularies were labeled as unknown token <unk>. Note that consecutive non-target vocabularies are consolidated into a single <unk> token. The labeling of the non-target vocabulary sequence was performed

Table 1: *CERs for the target vocabulary: katakana and numerals. The single task and the proposed multi-task show both results trained from scratch and fine-tuned using the ASR model with full vocabulary. Overall, the proposed method shows lower CERs. It can also be seen that the single task is difficult to be trained from scratch.*

Model	λ	CER (target: katakana)↓				CER (target: numeral)↓			
		eval1	eval2	eval3	all	eval1	eval2	eval3	all
Training from scratch (random initialization)									
ASR + Filtering	-	11.55	10.71	**9.22**	10.77	4.6	5.02	5.04	4.87
Single task	-	Failed in training (loss diverged)							
Multi task	0.25	11.13	10.71	10.23	10.79	**3.31**	5.82	3.78	4.65
	0.5	**9.59**	**10.2**	10.29	**9.95**	7.05	7.66	5.88	7.22
	0.75	15.7	14.62	13.35	14.82	**3.31**	**5.01**	**3.36**	**4.18**
Fine-tuning with pre-trained ASR with full vocabulary									
Single task	-	10.23	11.16	10.89	10.71	5.04	6.23	5.88	5.75
	0.25	10.4	10.55	11.61	10.7	3.52	5.52	5.04	4.71
Multi task	0.5	8.77	10.42	10.23	9.68	3.31	**5.41**	4.62	**4.55**
	0.75	**8.49**	**10.16**	**10.17**	**9.45**	**3.02**	5.92	**3.78**	4.6

Table 2: *Insertion error rate (Ins) for utterances that do not contain katakana when the target vocabulary is katakana. eval1, eval2, and all show that the proposed method has lower Ins. The proposed multi-task network is robust for non-target vocabulary.*

Model	λ	Ins for non-target vocabulary (%)↓			
		eval1	eval2	eval3	all
Filtering	-	0.84	1.8	**1.01**	1.21
Single task (finetune)	-	0.6	1.24	**1.01**	0.96
Multi task (scratch)	0.5	0.48	0.68	1.38	0.89
Multi task (finetune)	0.75	**0.12**	**0.56**	1.57	**0.82**

by the reverse operation: the target vocabulary was labeled as <unk>, and the non-target vocabulary was left as teacher sequences.

We used the character error rate (CER) of the target vocabulary characters only, as the evaluation metric for each evaluation set. We also calculated the insertion error rate (Ins) for utterances that *do not* contain the target vocabulary.

4.2. Network configurations

We used 80-dimensional Mel-spectrogram features as input features. SpecAugment [33] was applied for all models.

The encoder layer, the encoder dimension, the convolution kernel size, and the number of attention heads were set to 18, 512, 31, and 8, respectively. We trained the models with the Adam optimizer [34] with $\beta_1 = 0.9$, $\beta_2 = 0.98$, $\epsilon = 10^{-6}$ and Noam learning rate scheduler from [35], with 1k warm-up steps and peak learning rate 0.05 / $\sqrt{d}$ where d is the model dimension in conformer encoder. All models were trained for 100 epochs, and fine-tuned models were performed for an additional 100 epochs using a pre-trained model of the full vocabulary which is used for the filtering approach.

4.3. Experimental results

Table 1 shows CERs for the target vocabulary with the full-vocabulary ASR[1]+filtering approach, the single-task approach, and the multi-task approach with varying weights of λ. The full-vocabulary ASR model used for filtering was also fine-tuned to obtain results for the single-task learning approach and multi-task learning approach. Overall, the proposed multitask learning approach showed lower CER in all experimental settings. In particular, a significant improvement was observed in the eval1 set, where the proposed approach showed a relative CER improvement of 27% for the katakana target and 34% for the numerals target, compared to filtering.

In the single-task learning approach learning from scratch was not feasible and fine-tuning was required, as discussed in Sections 2 and 3.2. This is because training from scratch using only target vocabulary does not allow for the use of information on non-target vocabulary contained in the training data.

In contrast, the multitask learning approach was found to be feasible for learning from scratch, although the CER showed fluctuations depending on the weight λ. Moreover, fine-tuning further improved the accuracy and helped to stabilize the training process with small fluctuations observed around the CERs. The best CER was observed when λ was set to 0.75. This is because the pre-trained model had already learned representations for the full vocabulary, and placing a larger weight on the target vocabulary could enhance the accuracy of target extraction.

Table 2 shows the insertion error rates (Ins) on utterances containing only non-target vocabulary. This can be seen as a false alarm rate which can measure the robustness of the system to non-target vocabulary. Despite some degradation in performance on eval3, the proposed multi-task training approach reduced the number of insertions by 86% on eval1, 69% on eval2, and 32% overall on the dataset. This suggests that the proposed approach was able to capture the differences between target and non-target vocabulary and improve the recognition of target vocabulary while maintaining robustness to non-target vocabulary.

5. Conclusions

In this paper, we proposed a new approach for target vocabulary recognition based on multi-task learning with decomposed teacher sequences. By utilizing information from both target and non-target vocabulary, our proposed method provides more stable training and more accurate recognition of target vocabulary. The experiments using the CSJ dataset demonstrate that the proposed method showed improved accuracy compared to the conventional methods such as target vocabulary filtering and single-task learning. We will explore the effectiveness of our approach in other languages in future work.

[1]The ASR model utilized in this approach demonstrates CERs of 4.9%, 3.5%, and 4.1% for eval1, eval2, and eval3, respectively, in the full vocabulary recognition, i.e. the conventional ASR task.

6. References

[1] A. Graves, "Sequence Transduction with Recurrent Neural Networks," in *Proc. ICML*, 2012.

[2] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. ICASSP*, 2013, pp. 6645–6649.

[3] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[4] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Proc. NeurIPS*, 2015, pp. 577–585.

[5] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Proc. NeurIPS*, 2020.

[6] N. Chen, S. Watanabe, J. Villalba, P. Żelasko, and N. Dehak, "Non-Autoregressive Transformer for Speech Recognition," *IEEE Signal Processing Letters*, vol. 28, pp. 121–125, 2021.

[7] Y. Higuchi, S. Watanabe, N. Chen, T. Ogawa, and T. Kobayashi, "Mask CTC: Non-Autoregressive End-to-End ASR with CTC and Mask Predict," in *Proc. Interspeech*, 2020, pp. 3655–3659.

[8] J. Lee and S. Watanabe, "Intermediate Loss Regularization for CTC-Based Speech Recognition," in *Proc. ICASSP*, 2021, pp. 6224–6228.

[9] J. Nozaki and T. Komatsu, "Relaxing the Conditional Independence Assumption of CTC-Based ASR by Conditioning on Intermediate Predictions," in *Proc. Interspeech*, 2021, pp. 3735–3739.

[10] Y. Higuchi, N. Chen, Y. Fujita, H. Inaguma, T. Komatsu, J. Lee, J. Nozaki, T. Wang, and S. Watanabe, "A comparative study on non-autoregressive modelings for speech-to-text generation," in *Proc. ASRU*, 2021, pp. 47–54.

[11] K. Rao, H. Sak, and R. Prabhavalkar, "Exploring architectures, data and units for streaming end-to-end speech recognition with RNN-transducer," in *Proc. ASRU 2017*, 2017, pp. 193–199.

[12] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented Transformer for Speech Recognition," in *Proc. Interspeech*, 2020, pp. 5036–5040.

[13] J. Guo *et al.*, "Efficient minimum word error rate training of RNN-transducer for end-to-end speech recognition," in *Proc. Interspeech*, 2020.

[14] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," in *Proc. ICASSP*. IEEE, 2016, pp. 4945–4949.

[15] L. Dong, S. Xu, and B. Xu, "Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition," in *Proc. ICASSP*. IEEE, 2018, pp. 5884–5888.

[16] 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, pp. 4960–4964.

[17] S. Watanabe, T. Hori, S. Kim, J. R. Hershey, and T. Hayashi, "Hybrid CTC/Attention Architecture for End-to-End Speech Recognition," *JSTSP*, vol. 11, no. 8, pp. 1240–1253, 2017.

[18] G. Chen, C. Parada, and G. Heigold, "Small-footprint keyword spotting using deep neural networks," in *Proc. ICASSP*. IEEE, 2014, pp. 4087–4091.

[19] T. N. Sainath and C. Parada, "Convolutional neural networks for small-footprint keyword spotting," 2015.

[20] X. Chen, S. Yin, D. Song, P. Ouyang, L. Liu, and S. Wei, "Small-footprint keyword spotting with graph convolutional network," 2019.

[21] K. R, V. K. Kurmi, V. Namboodiri, and C. V. Jawahar, "Generalized Keyword Spotting using ASR embeddings," in *Proc. Interspeech*, 2022, pp. 126–130.

[22] B. Kim, S. Yang, I. Chung, and S. Chang, "Dummy Prototypical Networks for Few-Shot Open-Set Keyword Spotting," in *Proc. Interspeech*, 2022, pp. 4621–4625.

[23] M. Radfar, A. Mouchtaris, and S. Kunzmann, "End-to-End Neural Transformer Based Spoken Language Understanding," in *Proc. Interspeech*, 2020, pp. 866–870.

[24] L. Lugosch, M. Ravanelli, P. Ignoto, V. S. Tomar, and Y. Bengio, "Speech Model Pre-Training for End-to-End Spoken Language Understanding," in *Proc. Interspeech*, 2019, pp. 814–818.

[25] S. wen Yang, P.-H. Chi, Y.-S. Chuang, C.-I. J. Lai, K. Lakhotia, Y. Y. Lin, A. T. Liu, J. Shi, X. Chang, G.-T. Lin, T.-H. Huang, W.-C. Tseng, K. tik Lee, D.-R. Liu, Z. Huang, S. Dong, S.-W. Li, S. Watanabe, A. Mohamed, and H. yi Lee, "SUPERB: Speech Processing Universal PERformance Benchmark," in *Proc. Interspeech*, 2021, pp. 1194–1198.

[26] M. Omachi, Y. Fujita, S. Watanabe, and M. Wiesner, "End-to-end ASR to jointly predict transcriptions and linguistic annotations," in *Proc. NACCL 2021*. Online: Association for Computational Linguistics, Jun. 2021, pp. 1861–1871. [Online]. Available: https://aclanthology.org/2021.naacl-main.149

[27] M. Shibatani, *The languages of Japan*. Cambridge University Press, 1990.

[28] E. H. Jorden, M. Noda, T. Kusumoto, and S. Soga, *Japanese: The spoken language*. Yale University Press New Haven, 1987, vol. 1.

[29] M. Shibatani and T. Kageyama, "Introduction to the handbooks of japanese language and linguistics," *Handbook of Japanese Phonetics and Phonology.*, 2015.

[30] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks," in *Proc. ICML*, 2006, p. 369–376.

[31] K. Maekawa, "Corpus of spontaneous japanese: Its design and evaluation," in *Proc. SSPR*, 2003.

[32] T. Kudo, K. Yamamoto, and Y. Matsumoto, "Applying conditional random fields to Japanese morphological analysis," in *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*. Barcelona, Spain: Association for Computational Linguistics, Jul. 2004, pp. 230–237. [Online]. Available: https://aclanthology.org/W04-3230

[33] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition," in *Proc. Interspeech*, 2019, pp. 2613–2617.

[34] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proc. ICLR*, 2015.

[35] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. NeurIPS*, 2017, p. 6000–6010.