



# End-to-End Transformer-Based Open-Vocabulary Keyword Spotting with Location-Guided Local Attention

Bo Wei<sup>1</sup>, Meirong Yang<sup>1</sup>, Tao Zhang<sup>1</sup>, Xiao Tang<sup>1</sup>, Xing Huang<sup>1</sup>,  
Kyuhong Kim<sup>2</sup>, Jaeyun Lee<sup>2</sup>, Kiho Cho<sup>2</sup>, Sung-Un Park<sup>2</sup>

<sup>1</sup>Samsung R&D Institute China Xi'an, China

<sup>2</sup>Samsung Advanced Institute of Technology, Republic of Korea

{bo.wei, mrong01.yang, t1.zhang, xiao1.tang, xing1.huang}@samsung.com,  
{kkyung.kim, jaeyun91.lee, kiho.cho, st.park}@samsung.com

## Abstract

Open-vocabulary keyword spotting (KWS) aims to detect arbitrary keywords from continuous speech, which allows users to define their personal keywords. In this paper, we propose a novel location guided end-to-end (E2E) keyword spotting system. Firstly, we predict endpoints of keyword in the entire speech based on attention mechanism. Secondly, we calculate the existence probability of keyword by fusing the located keyword speech segment and text with local attention. The results on Librispeech dataset and Google speech commands dataset show our proposed method significantly outperforms the baseline method and the latest small-footprint E2E KWS method.

**Index Terms:** Open-vocabulary, keyword spotting, end-to-end, keyword location, local attention

## 1. Introduction

With the growing popularity of voice control in intelligent devices, the need for high-performance keyword spotting methods becomes increasingly important. A common use is to activate devices by a pre-defined keyword. For example, Google's voice search uses the phrase "Okay Google" and Apple's conversational assistant uses the phrase "Hey Siri". However, this kind of operation is not personal, and the device may be activated by other speakers who use the same phrase. Thus, open-vocabulary KWS methods are proposed in recent years.

Most previous works use acoustic models (AM) to encode the audio speech into feature embeddings [1, 2, 3] or phonetic posteriorgram [4, 5, 6, 7, 8], and then decide whether there exists keyword based on some similarity comparison methods or a variety of search and decoding methods. Such as in [1], the LSTM hidden activations are extracted as speech embeddings and compared with Cosine distance, while in [4, 5], connectionist temporal classification (CTC) based confidence scores are introduced to determine the existence of keyword. The training objective of such two-stage methods may not exactly match the objective of KWS, making the whole system sub-optimal.

Therefore, it is desirable to design an end-to-end KWS system, and several recent works have made valuable explorations. [9] proposes an ASR-free E2E system, which produces audio embedding and keyword embedding by the acoustic encoder and keyword encoder respectively, and then merges them into multilayer perceptron (MLP) for keyword existence prediction. Similar E2E structures are also proposed in [10] and [11]. These three works use the fixed-length speech embedding where the temporal dimension is lost. Whereas, the temporal dimension is vital for KWS, argued in the recent work [12], in which, speech embeddings of all time steps are produced by a GRU encoder,

and the keyword embedding is merged into the speech embedding of each time step, to predict the existence probability of keyword along the temporal dimension.

In this paper, we propose a location guided E2E KWS system. We preserve the temporal dimension of speech as well. More importantly, we argue that the keyword speech segment can provide the most crucial information for keyword identification. Thus, we predict the existence probability of keyword by localizing keyword from the entire speech and classifying keyword through fusing the located keyword speech segment and text with local attention. We compare our methods with two typical open-vocabulary KWS methods of the acoustic model & CTC decoding baseline and a vanilla E2E KWS method. The results demonstrate the advantages of our method on both accuracy and generalization. Overall, our contributions can be summarized as follows:

- We introduce a Transformer [13] based keyword localization method without requiring frame-wise alignment corpus for open-vocabulary KWS task.
- We propose a location guided local attention method to concentrate on keyword related information, which greatly improves the accuracy of keyword spotting.
- An E2E network is trained with multi-task learning fashion in which an alignment loss is proposed to discriminate the positive and negative samples more effectively.

The rest of this paper is organized as follows: Section 2 describes the detailed structures of KWS network. The experiments are described in detail in Section 3, followed by the results and analysis in Section 4. Finally, we conclude our work in Section 5.

## 2. Methods

In this section we first present the acoustic model & CTC decoding based KWS as the baseline and then extend the baseline to a vanilla E2E KWS method by adding the keyword encoder and classification module. Finally, we propose our location guided E2E KWS method by introducing three major improvements to vanilla E2E KWS, including keyword endpoints localization, location guided local attention, and the alignment loss. To make a competitive baseline and fair comparison, all networks are based on the self-attention architecture [13, 14].

### 2.1. Acoustic model & CTC decoding

As shown in Figure 1 (a), the Transformer based acoustic model is trained with CTC loss and output probability of the target se-

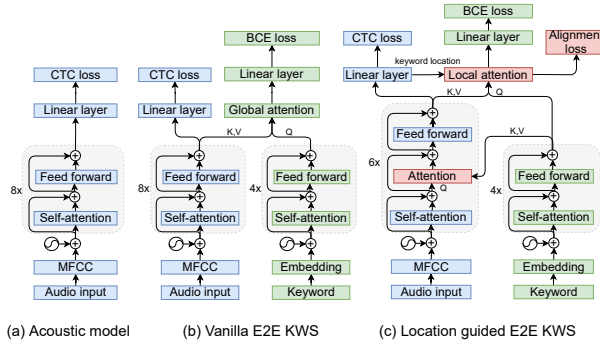


Figure 1: Structures of KWS networks presented in this work.

quence. In the decoding step, we follow the CTC-based keyword confidence score calculation and search strategy proposed in [5].

## 2.2. Vanilla E2E KWS

Although CTC has been used successfully in many previous works in the context of ASR, the two-stage KWS method makes the whole system sub-optimal. The vanilla E2E KWS network improves the baseline approach by adding the keyword encoder and the attention based classifier, as shown in Figure 1 (b).

The attention calculation used in this paper is formalized as

$$\mathbf{A} = \text{softmax}\left(\frac{f(\mathbf{Q}, \mathbf{K})}{\sqrt{d_k}}\right) \quad (1)$$

$$\mathbf{Z} = \mathbf{A}\mathbf{V} \quad (2)$$

where  $\mathbf{Q}$  is query matrix,  $\mathbf{K}$  is key matrix,  $\mathbf{V}$  is value matrix,  $f(\cdot)$  refers to a similarity function such as dot-product,  $d_k$  is the dimension of the row vector of  $\mathbf{K}$ ,  $\mathbf{A}$  is the attention weights, and  $\mathbf{Z}$  is the attention context output.

Here, speech embedding is extracted by acoustic model and fed into the attention layer as  $\mathbf{K}$  and  $\mathbf{V}$  matrixes. Similarly, keyword embedding is extracted by the keyword encoder and fed into the attention layer as  $\mathbf{Q}$  matrix. Then we use the first vector of the attention output as the classification vector (inspired by BERT [15]). After a linear layer, the existence probability of the keyword in the speech utterance is output.

Binary cross entropy (BCE) loss is used to jointly train the E2E network with CTC loss. We create positive examples where the keyword is present in audio, as well as negative examples where the keyword is absent in audio.

## 2.3. Location guided E2E KWS

In vanilla E2E KWS architecture, the audio encoder extracts speech embedding of the whole utterance. We argue that the speech segment containing keyword contributes most to the detection of the existence of keyword. Other irrelevant parts of the speech may introduce interference information. Thus, we predict the keyword location first, and then classify whether there exists the keyword or not with the location guided local attention, as shown in Figure 1 (c) (see more details in Figure 2). Specifically, we extend the vanilla E2E KWS from the following aspects.

### 2.3.1. Keyword endpoints localization

Inspired by the work of [8], for positive samples, we add the “<start>” and “<end>” markers before and after the keyword in

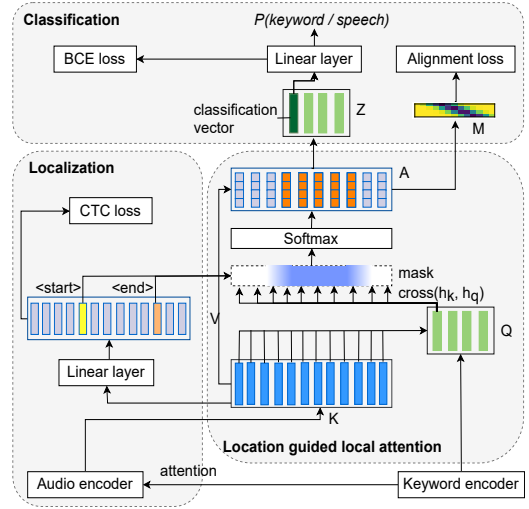


Figure 2: Details of the proposed location guided KWS method.

the transcribed text of the utterance. For example, for the utterance “Hi Freeman call my wife”, if we sample “Freeman” as the keyword, we would modify the target transcript text as “Hi <start> Freeman <end> call my wife”. In this way, the two markers are served as two special phoneme tokens and the corresponding probabilities could be output by audio encoder.

We then add a multi-head cross-attention layer after each self-attention layer in the audio encoder, and inject the keyword embedding as the  $\mathbf{K}$  and  $\mathbf{V}$  matrix (see formula (2)). In essence, we build a Transformer encoder/decoder architecture, where the Transformer encoder is used as keyword encoder, and the Transformer decoder is used as audio encoder. Thus the temporal dimension of audio embeddings can be kept after the cross attention, and the audio embeddings are biased towards the keyword.

The final audio embedding extracted by audio encoder is then fed into a linear layer to predict the probability of phoneme tokens for each time step. To simplify the problem, we assume that each sentence contains at most one keyword. Thus the positions of keyword endpoints are defined as

$$\text{position}(a) = \arg \max_t p_t^a \quad (3)$$

where  $a \in \{\langle \text{start} \rangle, \langle \text{end} \rangle\}$ ,  $t$  refers to the time step, and  $p_t^a$  means the probability of  $a$  occurs at time  $t$ .

### 2.3.2. Location guided local attention

Since we have got the keyword endpoints, we can enable the keyword embedding to focus only on audio embedding within the two endpoints. We achieve the local attention by adding a soft mask on the attention map before softmax operator. The mask is defined as

$$\text{mask}(t) = \begin{cases} \text{Gaussian}(s, \sigma), & t \leq s \\ 1, & s < t < e \\ \text{Gaussian}(e, \sigma), & t \geq e \end{cases} \quad (4)$$

where  $t$  is the time step of audio embedding,  $s$  and  $e$  refer to the start and end positions of the keyword.  $\sigma$  controls the softening level of the keyword boundary, and it is set to 0.5. When the value of  $\text{mask}(t)$  is lower than a threshold (0.01), we reset it to  $-1e9$ .

### 2.3.3. Alignment loss & multi-task learning

Because the time sequence of audio speech and the corresponding transcribed text is monotonous, we can observe a diagonal pattern in the attention map of the keyword and the speech segment containing the keyword. Based on this observation, we introduce alignment loss to constrain the alignment of keyword and speech with the local attention. The alignment loss is an extension of the guided attention loss [16], and we adapt it to discriminate positive and negative samples, which is defined as follows

$$L_A = \begin{cases} \text{mean}(\text{sum}(\mathbf{A} \otimes \mathbf{M}, \text{dim} = 1)), & \text{positive} \\ 1 - \text{mean}(\text{sum}(\mathbf{A} \otimes \mathbf{M}, \text{dim} = 1)), & \text{negative} \end{cases} \quad (5)$$

where  $\mathbf{A}$  is the local attention weights with shape  $w \times h$ ,  $\otimes$  refers to the element-wise multiplication,  $\mathbf{M}$  is a mask matrix which has a diagonal pattern with Gaussian distribution

$$M(i, j) = 1 - e^{-\frac{(\frac{i}{w} - \frac{j}{h})^2}{2\sigma^2}} \quad (6)$$

Overall, the proposed location guided E2E KWS network is a multi-task jointly training network, and the total loss is as follows

$$L_{\text{total}} = \lambda_1 L_{\text{CTC}} + \lambda_2 L_{\text{BCE}} + \lambda_3 L_A \quad (7)$$

where  $\lambda_1$ ,  $\lambda_2$ , and  $\lambda_3$  are the weights of the corresponding loss.

## 3. Experimental setup

### 3.1. Training dataset

We train the three KWS models on the training subsets of Librispeech [17] corpus, involving nearly 960 hours of speech audio. In order to improve environmental robustness, we add varying degrees of MUSAN [18] noise with SNRs ranging from 0 to 20dB, and reverberation from RIRS [19] respectively. The ratio of the original audio, noisy data, and reverberation data keeps 2:1:1. As to E2E-KWS training, we randomly select a word as the keyword in the transcribed text of the utterance to generate a positive sample, and randomly select a word in the lexicon not in the transcribed text of the utterance to generate a negative sample. The number of the positive and negative samples are kept the same. For each keyword, we sample the word containing 4 phones at least. Specifically, we further process the positive samples of the proposed location guided E2E KWS according to the methods described in Section 2.3.1.

### 3.2. Model details

The input acoustic signal is represented with 40-dimensional Mel-Frequency Cepstrum Coefficient (MFCC) features computed with a 25ms window and a 10ms frame-shift. Then SpecAugment [20] is applied to 5% of training data with default parameters. At last, we stack five consecutive frames and preserve one every three stacked frames [21] as input to the encoder.

For the baseline network, we use 8 Transformer self-attention blocks, each with 8 self-attention heads, 256-dimensional hidden size, and 1,024 dimensions for feed-forward layer. For the vanilla E2E KWS network, we add the keyword encoder which includes 4 Transformer self-attention blocks. For our location guided E2E KWS network, we reduce the number of the self-attention blocks of audio encoder to 6, and add a cross-attention layer to each block. Note that during inference, we only use the audio encoder and the classifier, and the keyword encoder is only used once when a new keyword is

enrolled. Thus for fair comparison, we keep the same number of parameters (about 6.3M) for the baseline network, the audio encoder and classifier parts of vanilla E2E KWS network as well as those two parts of the proposed location guided E2E KWS network.

Each of the models presented in this paper is trained on 1 GPU for 20 epochs with mini-batch of 16. The learning rate is set to 0.001 initially and warmed up at the first 4,000 steps. The Adam optimizer is used with  $\beta_1=0.9$  and  $\beta_2=0.98$ .

### 3.3. Evaluation

We evaluate our models on Librispeech dataset and Google speech commands dataset [22]. For Librispeech dataset, we construct about 7,700 pairs of positive and negative samples from the test subsets using the same method as the training data. For Google speech commands dataset, we follow the usual evaluation procedure [23] to adopt all audio clips of 10 keywords ("yes", "no", "up", "down", "left", "right", "on", "off", "stop", and "go") as the positive samples. And for each keyword, the same number of negative samples are randomly selected from other keywords. We evaluate the performance in terms of the receiver operating characteristic (ROC) curve and F1 score. The ROC curve is constructed by sweeping a threshold over all possible confidence values and plotting false reject (FR) rates against false accept (FA) rates. F1 score is defined as the harmonic mean of the precision and recall, and we use the value that maximizes the F1 score as the threshold.

## 4. Results

### 4.1. Comparison with baseline

Figure 3 (a) shows the ROC curves of performance comparison on Librispeech dataset for all proposed improvements upon the acoustic model & CTC decoding KWS baseline. As can be seen, E2E methods get much better results than acoustic model & CTC decoding baseline method, with large area under curve (AUC) decrease. Comparing to the vanilla E2E KWS network, our proposed location guided E2E KWS network reduces FRRs furthermore. Multi-task learning with alignment loss contributes slight performance improvement when FAR is higher than 0.025.

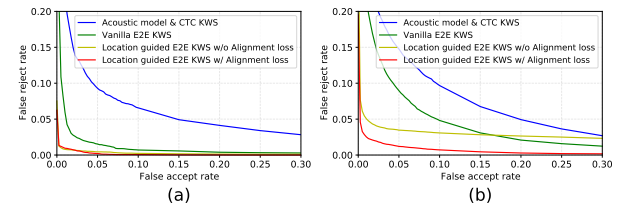


Figure 3: ROC curves of three KWS methods on (a) Librispeech dataset, and (b) Google speech commands dataset.

To evaluate the generalization ability of our proposed methods, we also test these methods on Google commands dataset, as shown in Figure 3 (b). Compared with Figure 3 (a), performances of all methods degrade due to the following reasons: 1) the Google commands dataset is not in the training dataset. 2) The ten keywords of Google commands dataset and the corresponding audio clips are very short (only 2-4 phones), which is different from the training data. Even though, there is just a slight performance drop for our proposed location guided

E2E KWS method with alignment loss during E2E training.

We plot representative examples of the attention weights of both positive and negative samples in Figure 4. Compared with the vanilla E2E KWS, our location guided E2E KWS produces more discriminative attention maps, with much more clear diagonal pattern in the positive attention map, and also much cleaner negative attention map.

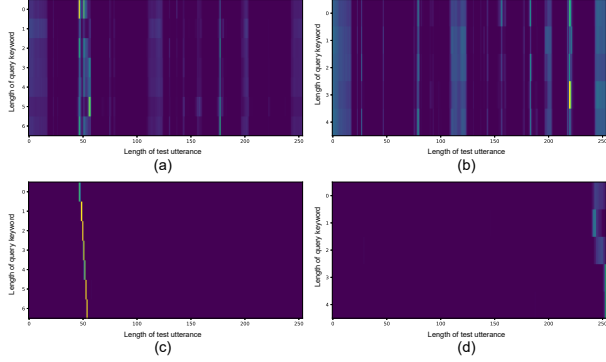


Figure 4: The visualization of attention maps for the test sample of “test-clean/4077/4077-13754-0009” in Librispeech dataset. (a) and (c) are attention maps of positive query for the keyword “plural” calculated by vanilla E2E KWS and location guided E2E KWS respectively. (b) and (d) are attention maps of negative query for the keyword “list” calculated by the two methods respectively.

## 4.2. Evaluation of keyword localization

To evaluate the localization accuracy, we test the localization method on TIMIT dataset [24], which provides frame-wise acoustic-phonetic labels. We use the test set of the TIMIT dataset with 1,680 utterances involved for evaluation. For each utterance, the word with more than 6 characters is selected as a keyword. A total of 3,880 test samples are generated.

For each test sample, the start and end positions of the keyword are predicted by our location guided E2E network. And then we use the central point of the two positions to represent the location of the keyword. We measured the location error between the predict results and the ground truth according to the number of phone offset. A phone offset contains average 1,282.3 sampled points in the .wav audio file, statistics from TIMIT corpus. Table 1 shows the location error for all 3,880 test samples. As can be seen, the location error of 95.39% of the samples is within one phone offset. This demonstrates acceptable accuracy of keyword localization.

Table 1: The distribution of keyword localization errors for TIMIT dataset

Number of phone offsets	% of samples
0 ~ 1	95.39%
1 ~ 2	1.03%
2 ~ 3	0.39%
> 3	3.19%

In order to intuitively illustrate the performance of localization, we show the localization results of four words in an utterance together in Figure 5. We can see that the predicted

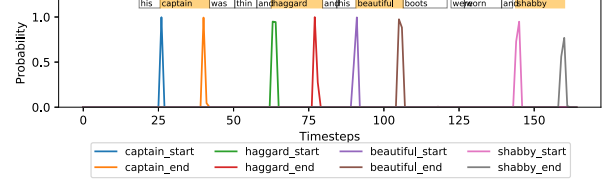


Figure 5: Visualization of keywords localization.

endpoints of each keyword almost exactly matches the ground truth.

## 4.3. Small-footprint model comparison

Finally, to facilitate the deployment of on-device KWS, we produce a small-footprint model (student) for the location guided E2E KWS network (teacher), following the distillation strategy proposed in TinyBERT [25]. The performance comparison with the teacher model on Google speech commands dataset is shown in Table 2.

Table 2: Distillation results of location guide E2E KWS

Model	Num. parameters	F1 score
Teacher model	6,300K	0.982
Student model	204K	0.919

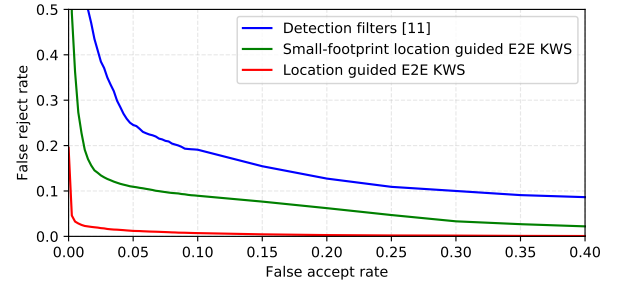


Figure 6: Small-footprint E2E KWS comparison.

One of the latest small-footprint open-vocabulary E2E KWS approaches, detection filters [11], contains 208.8K parameters. We compare it with our distilled model on Google speech commands dataset. As shown in Figure 6, our small-footprint model achieves much better performance than detection filters.

## 5. Conclusion

In this work, we develop a location guided end-to-end keyword spotting system based on Transformer network jointly training keyword location and keyword detection tasks. We propose a novel keyword location technique based on the cross attention of speech information and keyword information. Furthermore, we apply a location guided local attention instead of global attention to focus on the speech segment containing keyword. In experimental evaluations, we find that our system performs significantly better than baseline methods and the latest small-footprint E2E method. In addition, our proposed method shows better generalization ability on unseen data.

## 6. References

- [1] G. Chen, C. Parada, and T. N. Sainath, "Query-by-example keyword spotting using long short-term memory networks," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, South Brisbane, QLD, Australia, 2015, pp. 5236–5240.
- [2] J. Hou, L. Xie, and Z. Fu, "Investigating neural network based query-by-example keyword spotting approach for personalized wake-up word detection in mandarin chinese," in *2016 10<sup>th</sup> International Symposium on Chinese Spoken Language Processing (ISCSLP)*, Tianjin, China, 2016, pp. 1–5.
- [3] Ram, D. Miculicich, L. Bourlard, and Hervé, "Cnn based query by example spoken term detection," in *INTERSPEECH 2018 – 19<sup>th</sup> Annual Conference of the International Speech Communication Association*, 2018, pp. 92–96.
- [4] L. Lugosch, S. Myer, and V. S. Tomar, "Donut: Ctc-based query-by-example keyword spotting," in *ArXiv Preprint ArXiv:1811.10736*, 2018.
- [5] T. Bluche, M. Primet, and T. Gisselbrecht, "Small-footprint open-vocabulary keyword spotting with quantized lstm networks," in *ArXiv Preprint ArXiv:2002.10851*, 2020.
- [6] Y. Zhuang, X. Chang, Y. Qian, and K. Yu, "Unrestricted vocabulary keyword spotting using lstm-ctc," in *INTERSPEECH 2016 – 17<sup>th</sup> Annual Conference of the International Speech Communication Association*, 2016, pp. 938–942.
- [7] B. Kim, M. Lee, J. Lee, Y. Kim, and K. Hwang, "Query-by-example on-device keyword spotting," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019, pp. 532–538.
- [8] 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)*, 2017, pp. 474–481.
- [9] K. Audhkhasi, A. Rosenberg, A. Sethy, B. Ramabhadran, and B. Kingsbury, "End-to-end asr-free keyword search from speech," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 4840–4844.
- [10] N. Sacchi, A. Nanchen, M. Jaggi, and M. Cernak, "Open vocabulary keyword spotting with audio and text embeddings," in *INTERSPEECH 2019 – 20<sup>th</sup> Annual Conference of the International Speech Communication Association*, 2019, pp. 3362–3366.
- [11] Bluche, Théodore, and T. Gisselbrecht, "Predicting detection filters for small footprint open-vocabulary keyword spotting," in *INTERSPEECH 2020 – 21<sup>th</sup> Annual Conference of the International Speech Communication Association*, 2020, pp. 2552–2556.
- [12] Z. Zhao and W. Zhang, "End-to-end keyword search based on attention and energy scorer for low resource languages," in *INTERSPEECH 2020 – 21<sup>th</sup> Annual Conference of the International Speech Communication Association*, 2020, pp. 2587–2591.
- [13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Łukasz Kaiser, and I. Polosukhin, "Attention is all you need," in *the 31<sup>th</sup> International Conference on Neural Information Processing Systems (NeurIPS)*, 2017, pp. 6000–6010.
- [14] S. Adya, V. Garg, S. Sigtia, P. Simha, and C. Dhir, "Hybrid transformer/ctc networks for hardware efficient voice triggering," in *INTERSPEECH 2020 – 21<sup>th</sup> Annual Conference of the International Speech Communication Association*, 2020, pp. 3351–3355.
- [15] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 4171–4186.
- [16] H. Tachibana, K. Uenoyama, and S. Aihara, "Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4784–4788.
- [17] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, South Brisbane, QLD, Australia, 2015, pp. 5206–5210.
- [18] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," in *arXiv preprint arXiv:1510.08484*, 2015.
- [19] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5220–5224.
- [20] D. Park, W. Chan, Y. Zhang, C. Chiu, B. Zoph, E. Cubuk, and Q. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," in *INTERSPEECH 2019 – 20<sup>th</sup> Annual Conference of the International Speech Communication Association*, 2019, pp. 2613–2617.
- [21] H. Sak, A. Senior, K. Rao, and F. Beaufays, "Fast and accurate recurrent neural network acoustic models for speech recognition," in *INTERSPEECH 2015 – 16<sup>th</sup> Annual Conference of the International Speech Communication Association*, 2015, pp. 1468–1472.
- [22] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," in *arXiv preprint arXiv:1804.03209*, 2018.
- [23] J. L. Raphael Tang, "Deep residual learning for small-footprint keyword spotting," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5484–5488.
- [24] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, "Darpa timit acoustic-phonetic continuous speech corpus cd-rom TIMIT," in *NIST Interagency/Internal Report (NISTIR) - 4930*, 1993.
- [25] X. Jiao, Y. Yin, L. Shang, X. Jiang, X. Chen, L. Li, F. Wang, and Q. Liu, "TinyBERT: Distilling BERT for natural language understanding," in *Findings of the Association for Computational Linguistics: EMNLP 2020*. Online: Association for Computational Linguistics, Nov. 2020, pp. 4163–4174.