Samsung Research

# CTC-aligned Audio-Text Embedding for Streaming Open-vocabulary Keyword Spotting

Sichen Jin, Youngmoon Jung, Seungjin Lee, Jaeyoung Roh, Changwoo Han, Hoonyoung Cho
*Samsung Research, South Korea*

## Abstract

- We introduce a novel approach for **streaming open-vocabulary** keyword spotting (KWS) with **text-based** keyword enrollment.
- The efficient audio-text embedding space is found combining:
  - CTC for aligning the speech input and the target keyword text and processing frame-wise information,
  - Multi-view loss for comparing the global information of the entire keyword.
- We also provide:
  - A training recipe that simultaneously learn the CTC alignment and the audio-text embedding,
  - An inference algorithm that saves the most likely path efficiently with Viterbi algorithm.
- As a result, our approach achieves competitive performance on the *LibriPhrase* dataset compared to the non-streaming methods with a mere **155K model parameters** and a **decoding algorithm with time complexity *O(U)***, where *U* is the length of the target keyword.

## Results

### Experiment Settings

- *LibriPhrase* Dataset
  - Train: 1-to-4-words phrases from *train-clean-100/360*
  - Test: phrases from *train-other-500* → *easy($LP_E$)* and *hard($LP_H$)* sets
- Input feature
  - 80-dim power-mel filterbank
  - window: 25ms, stride: 10ms

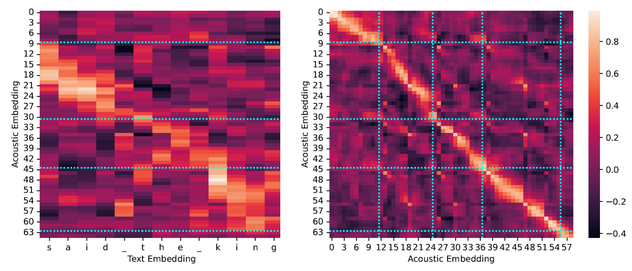
### Results

- Our system
  - Outperformed the non-streaming solutions on $LP_E$
  - Showed competitive results on $LP_H$
  - With merely 155K parameters and time complexity O(U), where U is the length of the keyword.
- We visualized the correlations between the embedding vectors of the positive pairs for the keyword “said the king”.
  a. The acoustic embedding and the text embedding.
  b. The acoustic embeddings of two different utterances.

* The blue lines denote the aligned results from CTC for the audio sections of <silence>, “said”, “the”, “king”, <silence>

| Model Specifications | |
|---|---|
| Acoustic | SeparableConv(k:12, d:96) x 12 layers |
| Text | BLSTM(256) x 2 layers |
| Embedding | 256 |
| Tokens | 28 |

| Method | #Params | EER (%) | | AUC (%) | |
|---|---|---|---|---|---|
| | | $\mathbf{LP_E}$ | $\mathbf{LP_H}$ | $\mathbf{LP_E}$ | $\mathbf{LP_H}$ |
| Attention [7] | ∼420K | 8.42 | 32.90 | 96.70 | 73.58 |
| DSP [9] | 3.7M | 7.36 | **23.36** | 97.83 | **84.21** |
| CTC | 147K | 9.11 | 32.37 | 96.76 | 73.95 |
| +character | 155K | 8.65 | 32.76 | 96.99 | 73.53 |
| +word | 155K | 7.05 | 31.62 | 97.97 | 75.12 |
| +phrase | 155K | **6.06** | 29.63 | **98.32** | 77.10 |




(a) *Audio-text embedding* (b) *Acoustic embedding*

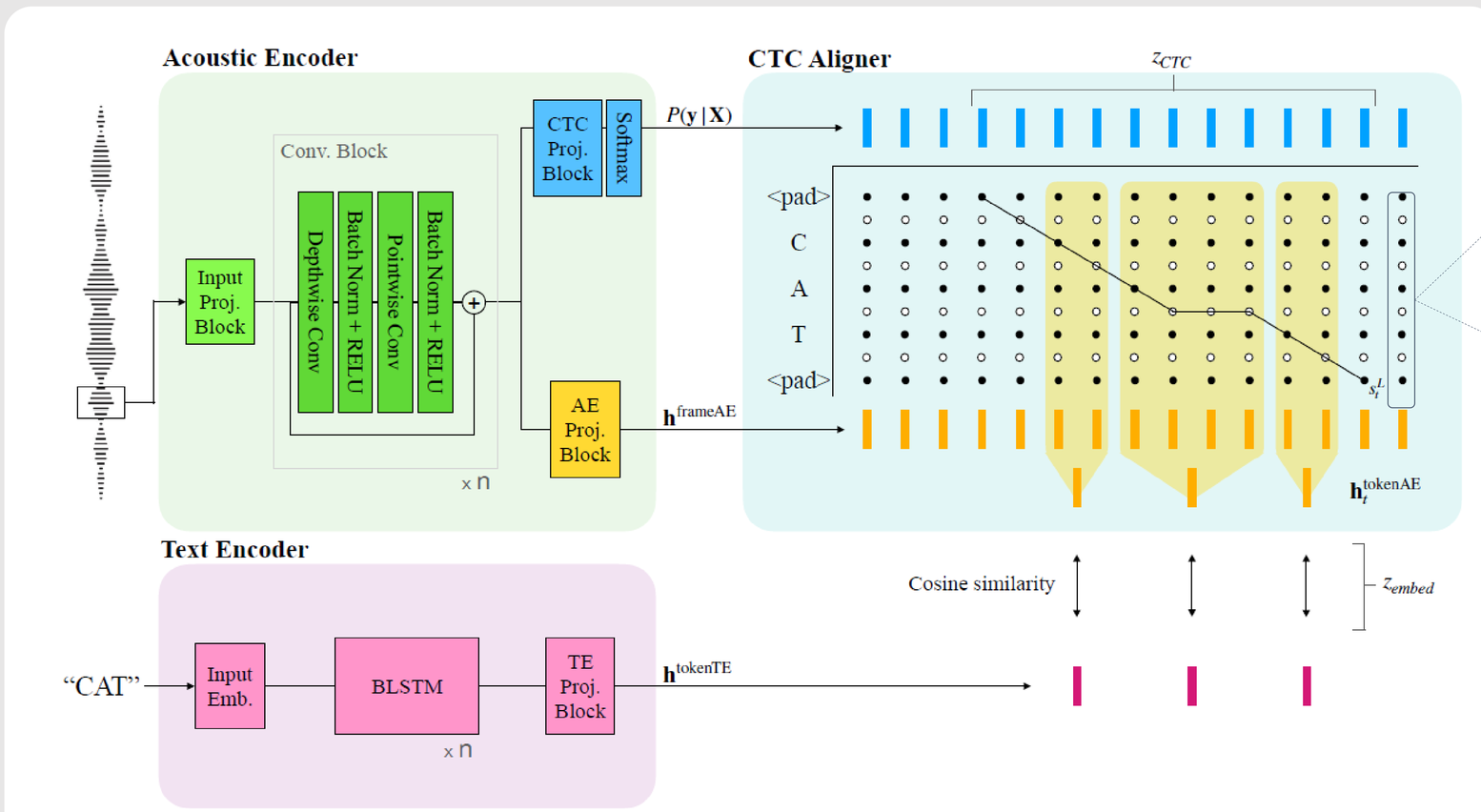



**Figure 1.** The depiction of the entire structure of the CTC-aligned Audio-Text keyword detector.

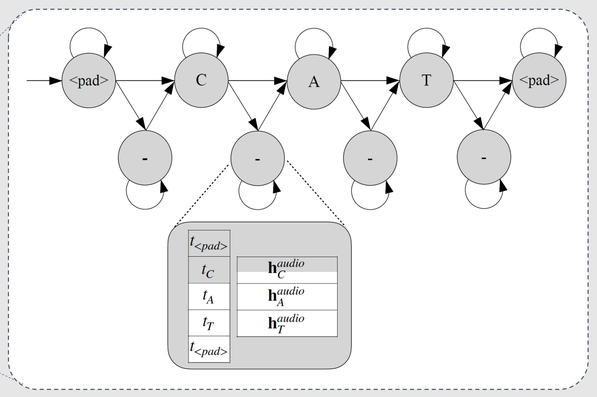



**Figure 2.** The decoding graph made for the keyword “cat”.

**Each state** in the graph saves **the most likely path** ending with the token it represents using **Viterbi algorithm:**

- CTC Score of the path,
- Transition timings of each non-blank token in the keyword,
- Accumulated embedding vectors for each non-blank token.

**Every time step, the states propagate** using the probability distribution over the vocabulary estimated from the CTC block.

**The last state** holds the most likely path for the whole keyword.

## Methods – Multi-task Learning

### CTC Loss

- The frame-wise prediction is learned with the CTC loss.
  - A special <blank> token is added match the length between audio and text.
  - The model finds the audio-text alignment without explicit training data.

$$p(\boldsymbol{l}|\boldsymbol{X}) = \sum_{\pi \in B^{-1}(\boldsymbol{l})} \prod_{t=1}^{T} y_{\pi_t}^t$$

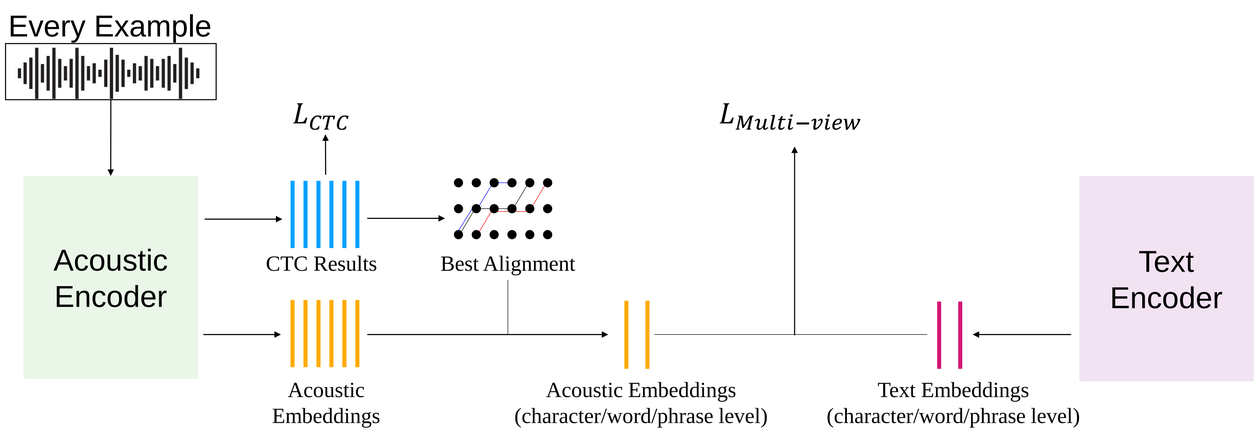
, where π is a possible alignment including the blank token, whose probability is the product of the probabilities of each token $\pi_t$ on the way.

### Multi-view Loss

- The cross-modal embedding space is learned with the multi-view loss.

$$L_{Multi-view} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{1}{\alpha}{}^{ELSE}_{j\in P_i}\alpha\left(\lambda - S(t_i, a_j)\right) + {}^{MSP}_{j\in N_i}\beta\left(S(a_i, t_k - \lambda)\right)\right)$$

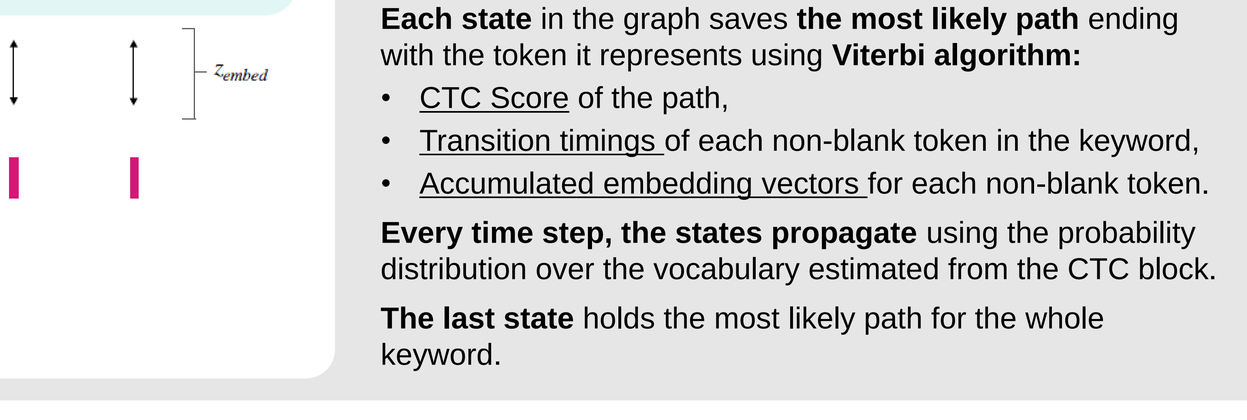
, where $P_i$, $N_i$, $S(\cdot,\cdot)$ are the positive sets, negative sets, and cosine similarity respectively. ELSE and MSP are Extended-LogSumExp and Mean-Softplus functions.




## Methods – Streaming Inference

- We use Viterbi algorithm to save the most likely path including its CTC score and the accumulated embedding vectors.
- The CTC score and the embedding score are linearly combined for the final score.

$$z = z_{CTC} + z_{embedding}$$




## Conclusion

- We found an efficient audio-text embedding space by aligning the audio input and the text input with CTC.
- We proposed a multi-task learning strategy that learns the alignment and the embedding space simultaneously.
- We achieved dynamic aligning between the two modalities for streaming spoken keyword detection at inference time.
- Experiments on the *LibriPhrase* dataset showed that our method gives competitive performance compared to non-streaming methods with much **smaller model size and time consumption**.