Efficient infusion of self-supervised representations in ASR

Darshan Prabhu, Sai Ganesh Mirishkar and Pankaj Wasnik

Sony Research India

The Motivation

Representations from self-supervised models is rich in **phoneme** and **word information** that are useful for **Automatic Speech Recognition**.

The Central Idea

In this work, we propose the **integration** of self-supervised and speech representations using methods like **framewise addition** and **cross-attention**.

Our Approach

Using **self-supervised** models directly as an encoder or a frontend in an ASR pipeline is **expensive**. If you have a **limited training budget**, an alternative approach is to use only the **representations** from these SSL models. Since the focus now is solely on the representations, they can be extracted beforehand as part of **pre-processing** (which can be easily done with parallel jobs). This one-time step eliminates the need for the SSL model during training, **significantly reducing training time** while sacrificing a minimal amount of efficiency.

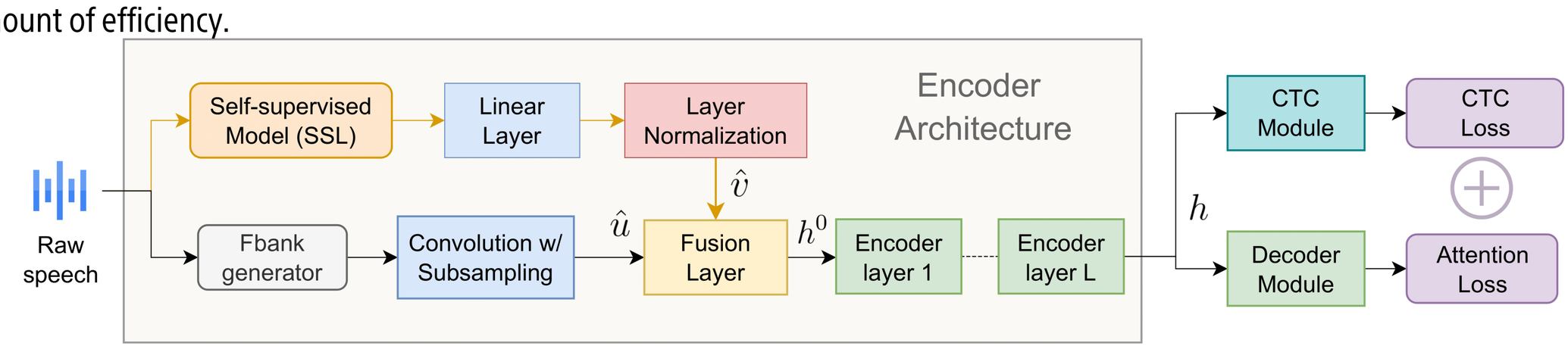


Figure 1: The overall architecture of our approach

In this work, we propose an **end-to-end ASR** architecture that efficiently **integrates self-supervised** model representations into the **speech encoder**. We accomplish this with a **fusion layer** that ranges from a simple **framewise addition** to a more complex **cross-attention** mechanism. Our complete architecture is illustrated in **Figure 1**. For the fusion layer, we have explored two straightforward approaches, as depicted in Figures 2 and 3 **Figure 2** provides an overview of Framewise addition-based fusion layer, which capitalizes on the linear relationship between the lengths of the representations. It uses **subsampling** to ensure that both representations are of equal length before performing **frame-level addition**. **Figure 3** demonstrates the utilization of **cross-attention** to merge the representations. This approach is **not dependent on the lengths**, making it a more potent replacement for framewise-addition.

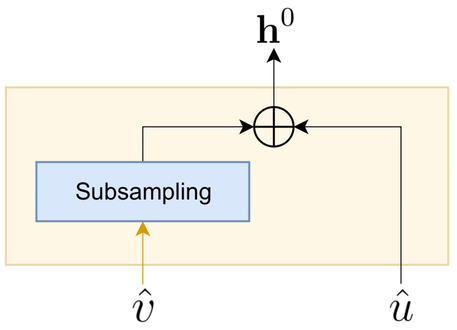


Figure 2: Using framewise addition as a fusion layer

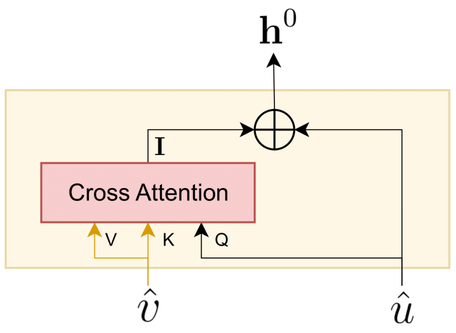


Figure 3: Using cross Attention as a fusion layer

Main Results

	Experiment	Dev Clean	Dev Other	Test Clean	Test Other
Baseline	Transformer	10.1	25.8	10.4	26.4
	Conformer	**7.9**	**21.4**	**8.4**	**22.0**
Our Method	Conf + Wav2Vec + SFA	5.9	16.2	6.4	16.4
	Conf + Wav2Vec + CA	5.9	16.0	6.3	16.3
	Conf + HuBERT + SFA	5.2	13.5	5.4	13.5
	Conf + HuBERT + CA	**5.1**	**13.0**	**5.4**	**13.3**

Comparison of performance of our system with baselines on Librispeech dataset. The SSL model is exposed to the ASR training data.

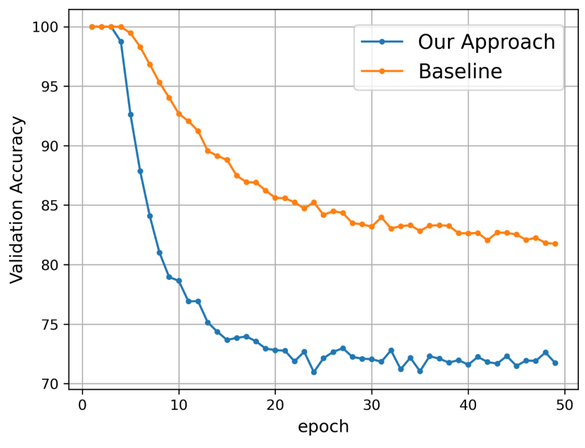
	Experiment	Dev	Test
Baseline	Conformer	**10.5**	**8.6**
Our Method	Conf + Wav2Vec Base + CA	9.6	9.3
	Conf + HuBERT Base + CA	9.4	8.9
	Conf + HuBERT Large + CA	**7.6**	**6.8**

Comparison of performance of our system with baselines on Tedlium2 dataset. The SSL model is not exposed to the ASR training data.

Key Findings

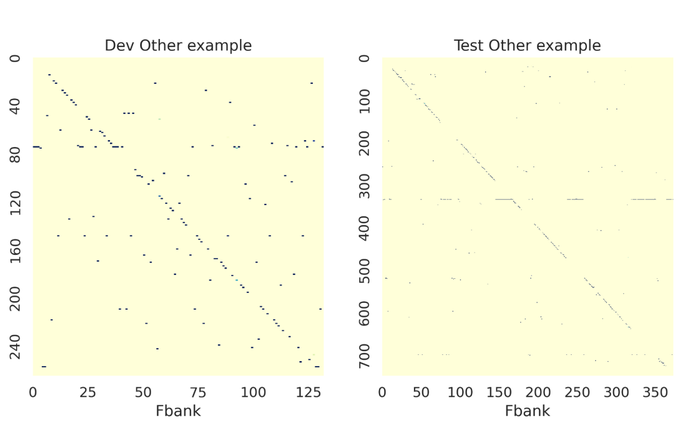
Faster Model Convergence

The utilization of the SSL representations leads to **rapid convergence** and we observe our model outperforming the baseline with only a few epochs of training.



Visualization of attention

Cross-attention based fusion learns to **predict the alignment** between the two representations, indicating local information is more important than global during fusion.



Number of Encoder layers

Even after reducing the number of encoder layers by 80%, we find that the model still performs much better despite having only half as many parameters and training time as the baseline model.

Experiment (# of params)	Dev Clean	Dev Other	Test Clean	Test Other
Baseline Conf. (30.6M)	**7.9**	**21.4**	**8.4**	**22.0**
Conf(E=12) + Hubert + CA (31.0 M)	5.1	13.0	5.4	13.3
Conf(E=8) + Hubert + CA (24.7 M)	**5.0**	12.9	**5.3**	13.0
Conf(E=4) + Hubert + CA (18.4 M)	5.4	12.8	5.7	**12.7**
Conf(E=2) + Hubert + CA (15.2 M)	5.5	**12.7**	5.5	12.8

For more details, you can email us at: saiganesh1.mirishkar@sony.com