AVFormer: Injecting Vision into Frozen Speech Models for Zero-Shot AV-ASR
Supplementary Material

Paul Hongsuck Seo Arsha Nagrani Cordelia Schmid
Google Research
{phseo, anagrani, cordelias}@google.com

Overview

In this supplementary material we provide additional ablations with varying the number of adapter layers and with more complex visual projectors in Section A and B, respectively. In Section C, we investigate the effects of iterative training. Then we supplement Table 2 of the main paper by providing results with more training dataset fractions in Section D, and show additional experiments replacing the RNN-T decoder with a cross-attentional transformer decoder in Section E. Finally, we present a failure case in Section F.

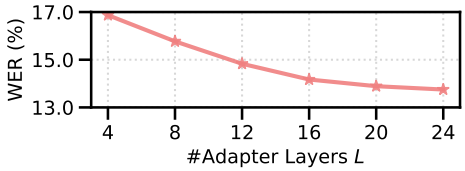
A. Number of Adapter Layers

Figure 1 shows the word error rate (WER) when varying the number of adapter layers from 4 to 24. Note that the number of the conformer blocks in BEST-RQ is 24 and therefore, the model with 24 adapters means that an adapter is added to every conformer block in the model. We also note that we add adapter layers to the last conformer blocks (before the decoder) when fewer than 24 layers are added to achieve the best performance.

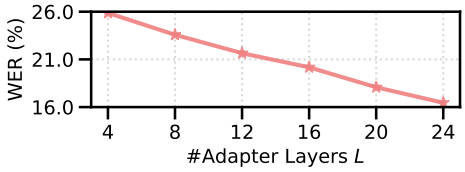
WER for How2 (Figure 1a) and VisSpeech (Figure 1b) monotonically decreases as we add more adapter layers to the model. For Ego4D (Figure 1c), the performance saturates at 20 layers. These results suggest that it is critical to inject an adapter into every conformer block.

B. More Complex Visual Projector

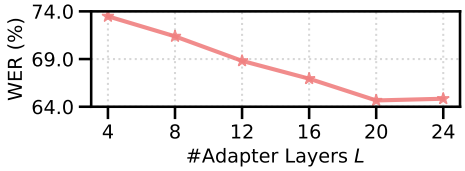
We also test a more complex visual projector in the form of a multi-layer perceptron (MLP) with varying number of layers (Table 1). The results consistently show on all three datasets that a single linear layer is sufficient for good performance (lower is better), and adding more layers makes a marginal impact (within error bars). Note that similar results are observed in [1] for prefix matching tasks.



(a) How2



(b) VisSpeech



(c) Ego4D

Figure 1. **Effect of the number of adapter layers.** Models are trained with 4 visual tokens using our curriculum learning strategy. Performance improve on all datasets as we increase the number of adapter layers. Lower WER is better.

# layers	How2	VisSpeech	Ego4D
1	**13.63 ± 0.10**	**16.39 ± 0.11**	**64.63 ± 0.79**
2	13.77 ± 0.09	16.47 ± 0.34	64.75 ± 0.81
3	13.93 ± 0.21	16.49 ± 0.14	65.20 ± 0.56
4	13.72 ± 0.11	16.49 ± 0.25	65.04 ± 0.50

Table 1. **Effect of the number of MLP layers in the visual projector.** ReLU is used as the intermediate activation function.

C. Iterative Training

In this section, we investigate iterative applications of our curriculum. We train our model for the second time, both with or without our proposed curriculum. The results in Figure 2 present performance degradation compared to

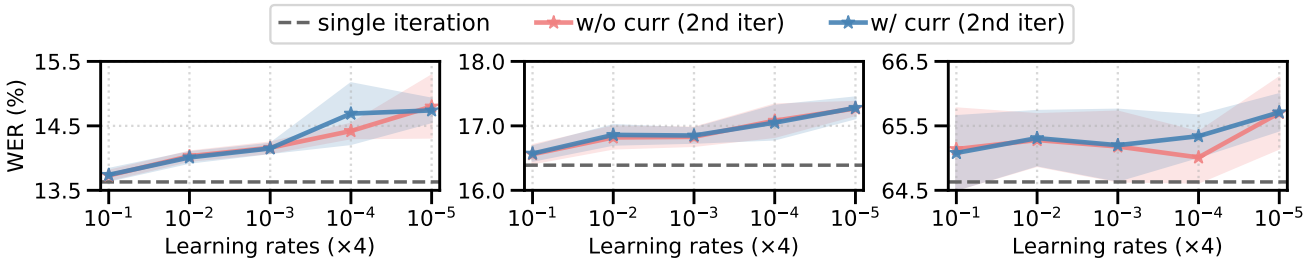


Figure 2. **Effects of iterative training on How2 and Ego4D.** Our model is trained for the second time with both or without our proposed curriculum using different learning rates.

Dataset Size	How2	VisSpeech	Ego4D
5%	13.69	16.60	64.75
10%	13.79	16.56	65.37
25%	13.60	16.57	64.29
50%	13.63	16.53	64.63
75%	13.66	16.69	65.11
100%	13.63	16.39	64.63

Table 2. **Effect of training dataset size.** Models are trained with 4 visual tokens using our curriculum strategy. Models are trained with varying fractions of HowTo100M. All scores are in WER% (lower is better). The results show that 5% of dataset is enough to achieve state-of-the-art performance.

our model with single iteration (gray dotted lines) in both cases on all three benchmarks. We observe that a larger learning rate increases the WER. We believe that this phenomenon is due to over-adaptation to HowTo100M.

D. Effect of Dataset Size

We extend the ablation presented in Table 2 of the main paper in Table 2. Due to the strong pretrained knowledge in BEST-RQ, we show that only a small fraction (5%) of the HowTo100M training dataset is enough to achieve comparable performance with training on the full dataset. This shows that our adapted model is extremely data efficient.

E. Autoregressive Decoder

Finally, we test our method with different decoders: an RNN-Transducer (RNN-T) and a transformer decoder using cross-attention (Cross-attention) introduced in [2]. RNN-T is the decoder used in the pretrained BEST-RQ model, we keep the weights frozen when training for AV-ASR. The cross-attention decoder performs autoregressive decoding while cross-attending to all input tokens. We stack 8 decoder transformer blocks; the weights are randomly initialized and tuned during the AV-ASR training.

Table 3 shows the results of these models on the four datasets (LibriSpeech and the three AV-ASR benchmarks).

Decoder	LibriSpeech	How2	VisSpeech	Ego4D
Cross-attention	13.79	16.67	20.21	70.47
RNN-T	4.40	13.63	16.39	64.63

Table 3. **Results with different decoders.** All scores are in WER% (lower is better). RNN-T represents that an RNN-T decoder is initialized with the pretrained BEST-RQ weights and frozen. Cross-attention means that we replace the original RNN-T decoder with a autoregressive transformer decoder using cross-attention on the input token embeddings. Results are reported on all three AV-ASR benchmarks as well as on LibriSpeech.

GT: and tie up both ends with a simple knot
Ours: and tie up both hands with a simple knot

Figure 3. **A failure example on VisSpeech.**

The cross-attention decoder performs worse than RNN-T on the three AV-ASR benchmarks, while performing *significantly* worse on Librispeech. Note that Cross-attention uses the entire set of input encodings for generating each output token whereas every output token is generated from a single input encoding with RNN-T. However, the results show that maintaining the pretrained decoding knowledge in RNN-T is more important than introducing larger flexibility in a finetuned decoder.

F. Failure Analysis

Figure 3 shows a failure case with an erroneous word 'hands' introduced by the visual input. However, we find this case very rare in our extensive qualitative exploration.

References

[1] Constantin Eichenberg, Sidney Black, Samuel Weinbach, Letitia Parcalabescu, and Anette Frank. Magma–multimodal augmentation of generative models through adapter-based finetuning. *arXiv preprint arXiv:2112.05253*, 2021. 1

[2] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2