PARAMETER-EFFICIENT CROSS-LANGUAGE TRANSFER LEARNING FOR A LANGUAGE-MODULAR AUDIOVISUAL SPEECH RECOGNITION

Zhengyang Li[*] *Thomas Graave*[*] *Jing Liu*[◦] *Timo Lohrenz*[*] *Siegfried Kunzmann*[◦] *Tim Fingscheidt*[*]

[*]Technische Universität Braunschweig
Institute for Communications Technology
38106 Braunschweig, Germany

[◦]Amazon Alexa AI
15203 Pittsburgh, PA, USA

ABSTRACT

In audiovisual speech recognition (AV-ASR), for many languages only few audiovisual data is available. Building upon an English model, in this work, we first apply and analyze various adapters for cross-language transfer learning to build a parameter-efficient and easy-to-extend AV-ASR in multiple languages. Fine-tuning only the bottleneck adapter with 4% of encoder's parameters and the decoder shows comparable performance to full fine-tuning in French and Spanish AV-ASR. Second, we investigate the effectiveness of various encoder components in cross-language transfer learning. Our proposed modular linguistic transfer learning approach excels the full fine-tuning method for German, French, and Spanish AV-ASR in almost all clean and noisy conditions (8/9). On low-resourced German AV data (13h), our proposed linguistic transfer learning achieves a 4.1% abs. WER reduction on average for clean and noisy speech, while fine-tuning only 50% of the encoder's parameters. Our code is at GitHub.[1]

Index Terms— audiovisual speech recognition, transfer learning, adapter, multi-lingual speech recognition

1. INTRODUCTION

Audiovisual speech recognition (AV-ASR) has been proven to be an effective approach in acoustically noisy and multi-talker conditions [1, 2, 3]. Compared to conventional ASR based on acoustics, AV-ASR leverages additional visual information, such as the movements of the speaker's lips and mouth region, to recognize the spoken utterances. The robustness of AV-ASR facilitates its deployment in cars or smart home devices, enabling more natural and accurate human-machine interactions.

Drawing upon the advancement of the all-attention-based transformer architecture in neural machine translation [4] and speech recognition [5, 6, 7], the transformer and its variant [8] have also been applied to AV-ASR [2, 9, 10]. The rapid improvement of AV-ASR in recent years also benefits from the availability of large public audiovisual corpora [9, 11, 12] and modern pre-training approaches through self-supervised learning (SSL) [13, 14, 15]. Shi et al. [15] proposed audiovisual hidden unit BERT (`AV-HuBERT`) as an encoder to learn a general audiovisual representation, which is pre-trained with SSL on unlabeled English datasets [11, 9, 12]. The `AV-HuBERT` encoder has shown its efficacy in various English downstream tasks such as AV-ASR [1, 15], automatic lip-reading [16, 17], speaker verification [18], and audiovisual speech enhancement [19]. To explain audiovisual features learned in `AV-HuBERT`, Pasad et al. [20] explore the correlation between intermediate outputs and features on different levels, such as acoustic and linguistic level. This respective explainability of pre-trained models is expected to help with adapting a pre-trained model in different downstream tasks.

The use of more English training data allowed further performance improvements of AV-ASR [21, 22, 23]. However, most languages are resource-constrained regarding public audiovisual data, which limits the deployment of AV-ASR in further languages. Recently, a few public multi-lingual audiovisual datasets [24, 25] as well as research on multi-lingual visual speech recognition [26] and multi-lingual AV-ASR [24] have emerged. Ma et al. [26] show that the model initialized by an English lip-reading task improves the lip-reading performance in low-resourced target languages compared to training in a target language from scratch. In their solution, an entire model needs to be stored for each language. Zadeh et al. [25] fine-tune the `AV-HuBERT` encoder pre-trained on English datasets by SSL on a multi-lingual audiovisual dataset. A universal model is applied for different languages during inference. However, this method has several drawbacks: First, the multi-lingual model needs more training time compared to monolingual AV-ASR. Second, the multi-lingual AV-ASR has to be re-trained when extending to a new language.

In natural language processing (NLP) tasks, large universal encoders pre-trained by SSL have shown state-of-the-art performance after fine-tuning in various downstream tasks [13]. In this context, adapters are introduced first in text-based NLP [27] as an alternative to fine-tuning. During training, only the light-weight adapters inserted into the pre-trained model and the prediction heads are fine-tuned in downstream tasks, while the pre-trained parameters in the encoders are frozen.

[1] **https://github.com/ifnspaml/Cross_Language_Transfer_Learning_AVASR.git**

979-8-3503-0689-7/23/$31.00

The memory requirements for training can thus be reduced. During inference, only the light-weight adapters and prediction heads for multiple tasks are stored in addition to a single shared pre-trained model to save storage. Inspired by this work, various adapter topologies have been proposed to improve the performance or to further reduce the number of trainable parameters [28, 29, 30]. With the wider deployment of large pre-trained speech models [14, 31, 32], adapters are also applied for different speech downstream tasks [33, 34, 35, 36, 37]. However, adapters so far have not been investigated in AV-ASR.

In this work, we show how to apply adapters for cross-language transfer learning in AV-ASR, which requires only a small amount of extra parameters instead of an entirely new fine-tuned `AV-HuBERT` *encoder*. Specifically, we are the first to insert light-weight adapters into the `AV-HuBERT` encoder, which is pre-trained on unlabeled English data. During fine-tuning in target languages, we only activate adapters as trainable parameters in the encoder for a parameter-efficient and modular cross-language transfer learning. Second, we investigate the utility of various adapters [27, 28, 30] for cross-language transfer learning in AV-ASR. We demonstrate that the adapters are able to be combined with the pre-trained `AV-HuBERT` to build a parameter-efficient AV-ASR, which can be easily extended to other languages in a modular fashion. Third, by investigating the contribution of different components in cross-language transfer learning, we introduce a parameter-efficient acoustic transfer learning and linguistic transfer learning.

The paper is structured as follows. In Section 2, we briefly introduce the baselines and a reference method. We revise various adapters in Section 3, while Section 4 presents our proposed acoustic and linguistic transfer learning approaches. The experimental setup is described in Section 5. Section 6 comprises experimental results and discussion on the multilingual MuAViC audiovisual speech recognition task [25], The paper is concluded in Section 7.

2. BASELINE AND REFERENCE METHODS

2.1. Baseline Methods

As shown in Fig. 1, the transformer encoder-decoder model for AV-ASR comprises an encoder (green block) and a transformer decoder. The encoder, which consists of an audiovisual frontend followed by positional encoding and N serial transformer encoder blocks, utilizes the image sequence $\mathbf{v}_1^T = (\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_T)$ and audio feature sequence $\mathbf{a}_1^{4T} = (\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_{4T})$ as input. Note that in our case the frame rate is 25 Hz (video) and 100 Hz (audio), causing the fourfold length $4T$ of the audio feature sequence. The autoregressive decoder leverages the encoded audiovisual representation and previous output token $c_{\ell-1}$ to predict the output token probability vector $\mathbf{P}_\ell$ of the current decoding step. The baseline model uses the original `AV-HuBERT` encoder without adapters [15]. We apply the following two training methods as our baseline approaches:

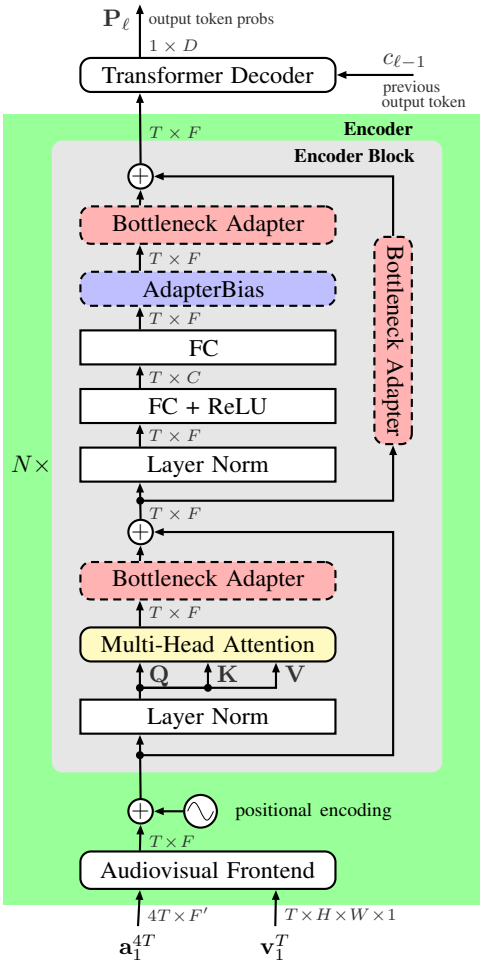
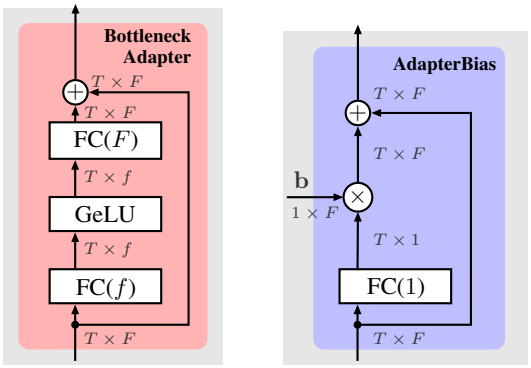


Fig. 1. **Transformer encoder-decoder model** with optional **adapters** used for audiovisual speech recognition **in inference**. The bottleneck adapter [27] and AdapterBias [29] are inserted into each transformer encoder block, which are presented in Fig. 2 (a) and Fig. 2 (b), respectively. Prefix-tuning [28] modifies the multi-head attention as detailed in Fig. 3. The skipping bottleneck adapter (again: Fig. 2 (a)) is a component of the Mix-and-Match adapter [30] only. During training, the encoder is frozen except the light-weight adapters.

Full fine-tuning: As the first baseline approach, the entire pre-trained encoder and the transformer decoder are trained on downstream tasks. The full fine-tuning is commonly regarded as the default method for transfer learning tasks. The `AV-HuBERT` encoder pre-trained on unlabeled English audiovisual data performs the AV-ASR task adaptation and the cross-language adaptation simultaneously when fine-tuning on audiovisual data in other languages.

Fixed encoder: The second baseline approach freezes the entire pre-trained encoder and trains only the transformer decoder. If not adapting the encoder to downstream tasks, a



(a) Bottleneck adapter (b) AdapterBias

Fig. 2. Adapters used **in inference** in Fig. 1: **(a) Bottleneck adapter** [27] **(b)** Fully connected layer with **AdapterBias** b.

performance loss can be expected.

2.2. Reference Method: Fine-Tuning Top Blocks

Training a model from scratch is time-consuming. In addition, an entire encoder needs to be stored for each downstream task. As the model learns structural features, and the features learned by lower blocks are similar across tasks, fine-tuning task-specific top (i.e., later) blocks is therefore a common approach as an alternative to full fine-tuning. For speech representation learning, the explainability of intermediate features has shown that the initial acoustic frontend and transformer encoder blocks learn acoustic features and the top transformer encoder blocks close to the decoder learn linguistic features [38, 20]. Fine-tuning top blocks can thus adapt linguistic patterns from the source language to the target language for cross-language transfer learning. In this work, we apply fine-tuning top transformer encoder blocks as reference method in parameter-efficient transfer learning for AV-ASR.

3. PARAMETER-EFFICIENT LANGUAGE-INDIVIDUAL ADAPTERS FOR AV-ASR

In this work, we propose to apply adapters for cross-language transfer learning to build a modular AV-ASR in multiple languages. The `AV-HuBERT` encoder pre-trained with SSL in English language is shared by various languages. In the encoder, only the light-weight adapters are updated during training and saved as language-specific parameters for each language. In this section, we show how to apply adapters in AV-ASR and revisit their topologies.

Bottleneck adapter: As shown in Fig. 2 (a), the bottleneck adapter [27] first converts the features from the original feature dimension F to the bottleneck dimension $f < F$, followed by a GeLU activation function, then projects the features back to dimension F. The bottleneck adapters, which are designed

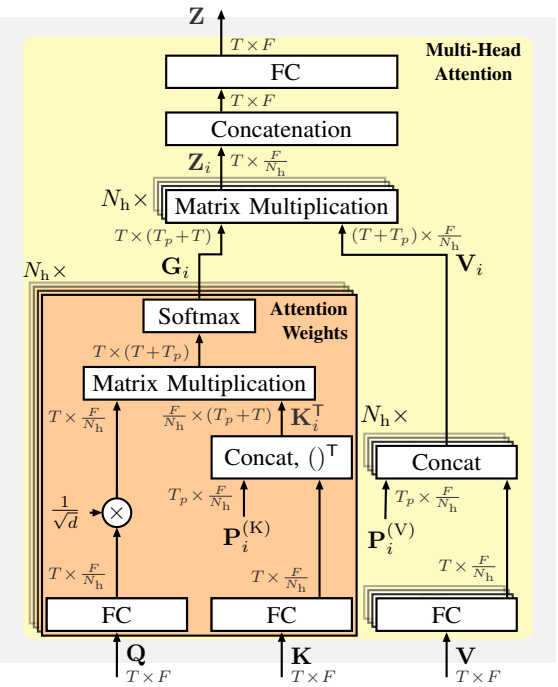


Fig. 3. **Multi-head attention** with **prefix-tuning** [28]. The prefix $\mathbf{P}_i^{(\mathrm{K})}$ of the key and the prefix $\mathbf{P}_i^{(\mathrm{V})}$ of the value are biases learned during **training**.

light-weight by setting a small bottleneck feature dimension f, are inserted into transformer blocks depicted as red blocks in Fig. 1.

Prefix tuning: Instead of inserting bottleneck adapters into transformer blocks, prefix tuning [28] modifies the multi-head (self-)attention (MHA) in each transformer encoder block as shown in Fig. 3. The query $\mathbf{Q}$, key $\mathbf{K}$, and value $\mathbf{V}$ inputs, which have a length T and a feature vector size F, are first processed by fully connected layers with parameter matrices $\mathbf{W}_i^{(\mathrm{Q})}, \mathbf{W}_i^{(\mathrm{K})}, \mathbf{W}_i^{(\mathrm{V})} \in \mathbb{R}^{F \times \frac{F}{N_\mathrm{h}}}$, respectively. The trainable prefix matrix $\mathbf{P}_i^{(\mathrm{K})}, \mathbf{P}_i^{(\mathrm{V})} \in \mathbb{R}^{T_p \times \frac{F}{N_\mathrm{h}}}$ with T_p rows are then concatenated with the projected key and value

$$\begin{aligned} \mathbf{K}_i &= \mathrm{Concat}(\mathbf{P}_i^{(\mathrm{K})}, \mathbf{K}\mathbf{W}_i^{(\mathrm{K})}) \in \mathbb{R}^{(T_p+T) \times \frac{F}{N_\mathrm{h}}} \\ \mathbf{V}_i &= \mathrm{Concat}(\mathbf{P}_i^{(\mathrm{V})}, \mathbf{V}\mathbf{W}_i^{(\mathrm{V})}) \in \mathbb{R}^{(T_p+T) \times \frac{F}{N_\mathrm{h}}} \end{aligned} \tag{1}$$

with the index $i \in \mathcal{N}_\mathrm{h} = \{1, \ldots, N_\mathrm{h}\}$ of the in total N_h attention heads. The MHA employs N_h attention heads

$$\mathbf{Z}_i(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \underbrace{\mathbf{softmax}\left(\frac{\mathbf{Q}\mathbf{W}_i^{(\mathrm{Q})} (\mathbf{K}_i)^\mathsf{T}}{\sqrt{d}}\right)}_{\substack{\text{attention weights} \\ =\mathbf{G}_i(\mathbf{Q},\mathbf{K},\mathbf{P}_i^{(\mathrm{K})})}} \cdot \mathbf{V}_i \in \mathbb{R}^{T \times \frac{F}{N_\mathrm{h}}} \tag{2}$$

The outputs $\mathbf{Z}_i$ of all N_h attention heads are concatenated and after a fully connected layer yield the output $\mathbf{Z} \in \mathbb{R}^{T \times F}$,

which has the same dimension as the input query, key, and value. The attention weights $\mathbf{G}_i \in [0,1]^{T\times(T_p+T)}$ depend on the trainable prefix $\mathbf{P}_i^{(\mathrm{K})}$, while the value $\mathbf{V}_i \in \mathbb{R}^{(T_p+T)\times\frac{F}{N_\mathrm{h}}}$ contains the trainable prefix $\mathbf{P}_i^{(\mathrm{V})}$, both helping to adapt the pre-trained MHA to downstream tasks.

Mix-and-Match: To leverage the benefits of the bottleneck adapters and prefix tuning, He et al. [30] proposed Mix-and-Match by leveraging prefixes to adapt the attention weights $\mathbf{G}_i$ and by replacing the skip connection around the second part of each transformer encoder block by a bottleneck adapter as shown in Fig. 1, for details see Fig. 2 (a).

AdapterBias: As a highly parameter-efficient adapter, AdapterBias [29] is inserted to each transformer encoder block (cf. Fig. 1). As shown in Fig. 2 (b), the features are first compressed from the feature dimension F to 1. The compressed features are multiplied with the trainable bias vector $\mathbf{b} \in \mathbb{R}^{1\times F}$ to be converted back to the original feature dimension F. The residual connection bypasses the AdapterBias.

4. EFFECTIVE AND PARAMETER-EFFICIENT TRAINING METHODS

In this work, we investigate parameter-efficient cross-language transfer learning methods for AV-ASR. In addition, we also explore the contribution of different components of the `AV-HuBERT` encoder during transfer learning. Recent advancements in the explainability of speech representations learned by SSL [20, 38, 39] show that large pre-trained models such as `Wav2vec 2.0` [31] and `AV-HuBERT` learn structural acoustic features in the AV frontend and early transformer encoder blocks and linguistic features in later blocks. Inspired by this finding, we design two types of transfer learning approaches for cross-language transfer learning in AV-ASR, i.e., an acoustic feature transfer learning method and a linguistic feature transfer learning method, exploring efficient and effective training approaches to reduce the number of trainable parameters and to improve performance simultaneously.

4.1. Acoustic Feature Transfer Learning

Pasad et al. [20] show that the features learned by the audiovisual frontend in `AV-HuBERT` have a large correlation to the acoustic features. Recent works [34, 40] adapt the acoustic feature or reprogram the audio input to improve the performance in downstream speech tasks. Motivated by their work, *we only introduce and fine-tune light-weight adapters in transformer encoder blocks and fine-tune the audiovisual frontend simultaneously* to build an *acoustic transfer learning method*.

4.2. Linguistic Feature Transfer Learning

Later transformer encoder blocks in `AV-HuBERT` are able to capture linguistic patterns [20]. Different languages can be acoustically similar but usually linguistically they differ a lot. In automatic speech recognition, linguistic features learned by later blocks are also proven to have more influence on the performance [39, 38]. Fine-tuning top (i.e., later) blocks of pre-trained models is thus common practice, but the resulting performance is usually worse or similar to the full fine-tuning. On the other hand, adapters show promising results by adapting features in different blocks with a small amount of trainable parameters. However, the resulting performance is insensitive to an increase of the number of adapter parameters beyond a certain point. As an effective and efficient *linguistic transfer learning method, we propose to combine the adapters and the method of fine-tuning top blocks to not only reduce the amount of trainable parameters but also to improve performance.*

5. EXPERIMENTAL SETUP

Databases and pre-processing: We fine-tune and evaluate models on the German (13 hours), French (179 hours), and Spanish (181 hours) subsets of the MuAViC dataset [25]. The MuAViC dataset [25] comprises audiovisual data in 8 languages collected from TED and TEDx talks on YouTube. Each language has `train`, `dev`, and `test` set. Note that the MuAViC dataset [25] only provides links of the YouTube videos to download, and some original links of the `train` sets were unavailable when we accessed the data (April 21st, 2023); we thus have less training data in this work. The video frame rate and the speech signal sample rate are 25 Hz and 16 kHz, respectively. In accordance with the pre-processing pipeline of the MuAViC dataset [25], we use 26-dimensional log-filterbank outputs as input audio features, which are extracted with a 25 ms window and a frame shift of 10 ms, resulting in 100 audio frames per second. Regarding video frames, we convert them to grayscale and crop them to a region of interest measuring 96×96 based on face alignment.

Fine-tuning for AV-ASR: The transformer encoder-decoder model is utilized in this work. The encoder is initialized by the pre-trained `large AV-HuBERT` [1] comprising 325M parameters in total. For a fair comparison, we apply the same decoder architecture as the baseline method [25] for all experiments. The decoder network consists of six transformer decoder blocks with 152M parameters. The outputs of the encoder-decoder architecture are subword tokens generated by `SentencePiece` [41] with a vocabulary size of 1000. The fine-tuning process is done using the `PyTorch`-based `fairseq` toolkit. We fine-tune the trainable parameters of the encoder-decoder model for 30k updates. The fine-tuning process uses batches of up to 1000 tokens. The learning rate is linearly increased to 0.001 for the first 10k updates, then linearly decreased to 0. We apply the same data augmentation as in the baseline method [25], where 25% of the training data is augmented with an SNR of 0dB noise chosen from the babble, music, natural noise, and second interfering talker conditions. There is no speaker overlap in babble noise and second interfering talker condition among different splits.

Evaluation in noisy environments: To add noise to our

Approach	# of trainable params in the encoder	WER (%)	
		dev	test
Baseline: [25]	325M	—	23.7
Baseline: [25], retrained	325M	23.8	25.5
Baseline: Fixed encoder	0	31.0	34.3
Reference method: Fine-tuning ...			
... top-1 block	13M	29.3	30.9
... top-4 blocks	50M	25.4	28.6
... top-8 blocks	101M	24.1	26.0
... top-12 blocks	151M	**23.4**	**25.8**
... AV frontend	14M	29.7	32.2
Recent adapters			
AdapterBias [29]	0.18M	27.6	30.0
Bottleneck adapter [27] ($f = 32$)	3M	25.9	27.2
Bottleneck adapter [27] ($f = 64$)	6M	24.9	27.2
Bottleneck adapter [27] ($f = 128$)	13M	**24.7**	**26.9**
Prefix tuning [28] ($T_p = 10$)	5M	27.4	30.5
Prefix tuning [28] ($T_p = 30$)	5M	27.4	30.0
Prefix tuning [28] ($T_p = 100$)	7M	27.6	29.6
Mix-and-Match [30] ($f = 32$)	7M	25.9	28.3
Mix-and-Match [30] ($f = 64$)	9M	25.6	27.7
Mix-and-Match [30] ($f = 128$)	12M	25.5	27.7

Table 1. **Ablation study:** WER (%) of AV-ASR on the **French** (FR) **dev** and **test** split in the MuAViC dataset. The center table segment reports on the reference method of fine-tuning different blocks of the encoder. The results of parameter-efficient training methods with various adapters are shown in the bottom segment. Best results of dev and test splits are in **bold** font, separately for the center and bottom table segment.

speech data, we follow the exact same procedure as detailed in [25, 1]. We generate babble noise by mixing utterances from 30 different speakers from the MUSAN dataset [42] where each speaker is used exclusively for either the train, dev, or the test split.

6. RESULTS AND DISCUSSION

To investigate the effectiveness of different parameter-efficient transfer learning methods and to find their suitable hyperparameters, we perform an ablation study as shown in Table 1. The AV-HuBERT pre-trained on English audiovisual data is adapted to French AV-ASR. The results are evaluated with clean speech on the French (FR) dev and test set of the MuAViC dataset [25].

The top table segment reports the results of baselines, where the baseline result reported by Anwar et al. [25] in the first row is in gray color, because the MuAViC dataset [25] we use is not directly comparable to the "original" one, see Section 5. Our retrained baseline in the second row fine-tunes the entire encoder with 325M parameters and the decoder, resulting in a 23.8% WER on the dev split and 25.5% on the test split. If the encoder is fixed (fixed encoder baseline), the performance significantly degrades compared to full fine-tuning, with a WER of 31.0% on the dev split and 34.4% on the test split, respectively.

The center table segment shows the results from fine-tuning different blocks of the encoder. We observe a performance improvement by fine-tuning more top transformer encoder blocks. By fine-tuning top-12 encoder blocks, the model already outperforms the retrained baseline on the dev split and is only slightly worse on the test split. In addition, fine-tuning the AV frontend is not as effective as fine-tuning the top-1 block even with 1M more trainable parameters. *This demonstrates that adapting linguistic features learned by top blocks is more advantageous than adapting acoustic features extracted by the AV frontend.*

The lower table segment shows various recent parameter-efficient adapters and an ablation study of their hyperparameters. All experiments with adapters outperform the fixed encoder baseline clearly on both dev and test splits, validating the effectiveness of fine-tuning light-weight adapters for cross-language transfer learning. The bottleneck adapter with the bottleneck dimension $f = 128$ shows best performance in the lower table segment, with comparable performance to full fine-tuning (retrained baseline), while fine-tuning only 4% of the encoder's parameters. The bottleneck adapter also excels fine-tuning the top-1 block with an absolute WER reduction of 4.6% on the dev split and 4.0% on the test split while fine-tuning a similar amount of parameters (13M). For the investigated adapters, the performance is insensitive to an increase in the number of adapter parameters beyond a certain point, which is also observed in text-based natural language processing tasks [27, 28, 30]. Based on the performance on the dev set, the bottleneck dimension $f = 128$ and prefix tuning with length $T_p = 30$ are applied in further experiments.

In Table 2, we apply and extend our proposed methods to other languages such as Spanish (ES) and German (DE). In addition, we also report the performance on clean speech, and at a signal-to-noise ratio (SNR) of 0dB and 10dB babble noise to evaluate the robustness of the AV-ASR model. Note that a lower SNR means a more noisy condition. The number of trainable parameters and all parameters in the encoder are presented for each method as well. In this table, the top segment shows the results of the baselines. The baseline [25] in the first row (i.e., cited numbers) is out of competition due to a number of broken YouTube links for training data, irreproducible noise access in training and inference, and irreproducible text normalization before computing the WER. The retrained baseline of full fine-tuning (2nd row) and the fixed encoder baseline (3rd row) are reported in the top table segment. Comparing the three languages, more training data leads to better performance. On the extremely low-resourced German (DE) split with only 13h audiovisual data, the retrained baseline only reaches a WER of 62.7% WER for clean speech on the test split. Expectedly, the performance degrades with the increase of the noise level (i.e., decrease of SNR) for all languages.

The bottom table segment first exhibits the results of the parameter-efficient methods such as fine-tuning top blocks and

Method	#params in the encoder		Word error rate (%)								
	trainable	all	DE (13h)			ES (181h)			FR (179h)		
			0dB	10dB	clean	0dB	10dB	clean	0dB	10dB	clean
Baseline: [25]	325M	325M	–	–	52.4	45.1	20.7	15.9	48.1	28.3	23.7
Baseline: [25], retrained	325M	325M	77.0	66.2	62.7	**44.9**	27.0	23.5	41.1	28.0	25.5
Baseline: Fixed encoder	0	325M	89.9	85.0	82.1	65.0	36.7	30.0	56.7	40.6	34.3
Parameter-efficient transfer learning methods:											
Fine-tuning top-1 block	13M	325M	87.3	82.4	79.9	62.0	34.9	28.9	52.4	35.9	30.9
Fine-tuning top-12 blocks	151M	325M	72.9	62.9	59.5	49.2	27.2	23.6	42.3	28.4	25.8
AdapterBias [29]	0.18M	325M	84.8	79.8	77.0	56.0	32.9	28.4	48.3	33.3	30.0
Bottleneck adapter [27] ($f=128$)	13M	338M	80.1	73.6	71.7	48.2	28.2	24.5	43.7	29.6	26.9
Prefix tuning [28] ($T_p=30$)	5M	330M	84.8	80.1	78.0	57.8	33.5	28.4	49.0	33.7	30.0
Mix-and-Match [30] ($T_p=30$, $f=128$)	12M	337M	80.3	73.8	71.6	51.4	30.0	26.1	45.1	30.7	27.7
Acoustic transfer learning:											
AV frontend + bottleneck adapter ($f=128$)	27M	338M	81.7	74.1	71.8	48.6	29.2	24.9	42.6	29.4	26.6
Linguistic transfer learning:											
Top-12 + bottleneck adapter ($f=128$)	164M	338M	**72.5**	**62.1**	**59.0**	45.1	**26.2**	**22.8**	**40.5**	**27.2**	**24.7**

Table 2. WER (%) on the **German** (DE), **Spanish** (ES), and **French** (FR) `test` split of the MuAViC dataset. Models are evaluated with **clean** speech, and at an SNR of **0dB** and **10dB babble** noise. Best results in the table are in **bold** font. The baseline [25] (i.e., cited numbers) is out of competition due to a number of broken YouTube links for training data, irreproducible noise access in training and inference, and irreproducible text normalization before computing the WER.

various adapters. Fine-tuning the top-1 block with 13M parameters and fine-tuning the top-12 blocks with 151M parameters are chosen as the reference methods. We apply the optimal hyperparameters of adapters found in the ablation study on French (FR) in Table 1 to German (DE) and Spanish (ES). All parameter-efficient methods perform better than the fixed encoder baseline with clean and noisy speech. The Spanish split (181h) and French split (179h) have similar data sizes, therefore the performance of parameter-efficient methods follows the same trend: Fine-tuning the top-12 blocks outperforms all adapters at a cost of fine-tuning 151M parameters, but it is still slightly worse than the full fine-tuning. The bottleneck adapter shows the best performance among the four investigated adapters, demonstrating a comparative performance to full fine-tuning while training only 4% of the encoder's parameters. However, on the low-resourced German split (13h), the bottleneck adapter is much worse than full fine-tuning (71.7% WER vs. 62.7% for clean speech). Interestingly, for low-resourced German, fine-tuning the top-12 blocks performs better than the retrained baseline with a 3.5% absolute WER reduction on average over clean and noisy speech.

Finally, the results of our proposed acoustic and linguistic transfer learning methods are presented in the bottom segment as well. The acoustic transfer learning, which fine-tunes the AV frontend and bottleneck adapters, doesn't show benefit compared to only the bottleneck adapter in 6 out of 9 test conditions. The linguistic transfer learning, which fine-tunes the top-12 blocks and bottleneck adapters, achieves the best performance in this table in almost all clean and noisy conditions (8/9) and even excels the retrained baseline by fine-tuning only 50% of the encoder's parameters. Especially on the low-resourced German split (13h), our proposed linguistic transfer learning excels full fine-tuning with a 4.1% absolute WER reduction on average over clean and noisy speech.

7. CONCLUSIONS

In this work, we first demonstrate how to apply various parameter-efficient adapters for cross-language transfer learning to achieve a modular and parameter-efficient audiovisual speech recognition (AV-ASR), which is easy to extend to further languages. Compared to the full fine-tuning, we show the effectiveness of the bottleneck adapter in French (FR) and Spanish (ES) with comparable performance, while only fine-tuning 4% of the encoder's parameters and the decoder. Second, we combine the bottleneck adapters and fine-tuning top encoder blocks to adapt the linguistic knowledge from a source language to a target language. Our proposed linguistic transfer learning outperforms the full fine-tuning for French, Spanish, and German AV-ASR in almost all clean and noisy conditions (8/9). Surprisingly, on the extremely low-resourced German data (13h), our proposed linguistic transfer learning excels the full fine-tuning with a 4.1% absolute WER reduction on average over clean speech, an SNR of 10dB and 0dB babble noise, while fine-tuning only 50% of the encoder's parameters.

8. ACKNOWLEDGMENTS

The research leading to these results has received funding from the Bundesministerium für Wirtschaft und Klimaschutz (BMWK) under funding code 01MK20011T (SPEAKER project) and Bundesministerium für Bildung und Forschung (BMBF) under funding code 03VP10991 (BesserLesen project).

9. REFERENCES

[1] B. Shi, W.-N. Hsu, K. Lakhotia, and A. Mohamed, "Robust Self-Supervised Audio-Visual Speech Recognition," *arXiv:2201.02184*, July 2022.

[2] P. Ma, S. Petridis, and M. Pantic, "End-To-End Audio-Visual Speech Recognition With Conformers," in *Proc. of ICASSP*, Toronto, ON, Canada, June 2021, pp. 7613–7617.

[3] S. Receveur, R. Weiss, and T. Fingscheidt, "Turbo Automatic Speech Recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 5, pp. 846–862, May 2016.

[4] 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. of NIPS*, Long Beach, CA, USA, Dec. 2017, pp. 1–11.

[5] L. Dong, S. Xu, and B. Xu, "Speech-Transformer: A No-Recurrence Sequence-to-Sequence Model for Speech Recognition," in *Proc. of ICASSP*, Calgary, AB, Canada, Apr. 2018, pp. 5884–5888.

[6] T. Lohrenz, Z. Li, and T. Fingscheidt, "Multi-Encoder Learning and Stream Fusion for Transformer-Based End-to-End Automatic Speech Recognition," in *Proc. of Interspeech*, Brno, Czech Republic, Sept. 2021, pp. 2846–2850.

[7] T. Lohrenz, P. Schwarz, Z. Li, and T. Fingscheidt, "Relaxed Attention: A Simple Method to Boost Performance of End-to-End Automatic Speech Recognition," in *Proc. of ASRU*, Cartagena, Colombia, Dec. 2021, pp. 177–184.

[8] 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. of Interspeech*, Shanghai, China, Oct. 2020, pp. 5036–5040.

[9] T. Afouras, J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman, "Deep Audio-Visual Speech Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–11, Dec. 2018, (early access).

[10] Z. Li, C. Liang, T. Lohrenz, M. Sach, B. Möller, and T. Fingscheidt, "An Efficient and Noise-Robust Audiovisual Encoder for Audiovisual Speech Recognition," in *Proc. of Interspeech*, Dublin, Ireland, Aug. 2023, pp. 1583–1587.

[11] J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman, "Lip Reading Sentences in the Wild," in *Proc. of CVPR*, Honolulu, HI, USA, July 2017, pp. 3444–3453.

[12] J. S. Chung, A. Nagrani, and A. Zisserman, "VoxCeleb2: Deep Speaker Recognition," in *Proc. of Interspeech*, Hyderabad, India, Sept. 2018, pp. 1086–1090.

[13] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding," in *Proc. of NAACL-HLT*, Minneapolis, MN, USA, June 2019, pp. 4171–4186.

[14] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, Oct. 2021.

[15] B. Shi, W.-N. Hsu, K. Lakhotia, and A. Mohamed, "Learning Audio-Visual Speech Representation by Masked Multimodal Cluster Prediction," in *Proc. of ICLR*, virtual, Apr. 2022, pp. 1–24.

[16] T. Lohrenz, B. Möller, Z. Li, and T. Fingscheidt, "Relaxed Attention for Transformer Models," in *Proc. of IJCNN*, Gold Coast, Australia, June 2023, pp. 1–10.

[17] Z. Li, T. Lohrenz, M. Dunkelberg, and T. Fingscheidt, "Transformer-Based Lip-Reading with Regularized Dropout and Relaxed Attention," in *Proc. of SLT*, Doha, Qatar, Jan. 2023, pp. 723–730.

[18] B. Shi, A. Mohamed, and W.-N. Hsu, "Learning Lip-Based Audio-Visual Speaker Embeddings with AV-HuBERT," in *Proc. of Interspeech*, Incheon, Korea, Sept. 2022, pp. 4785–4789.

[19] W.-N. Hsu, T. Remez, B. Shi, J. Donley, and Y. Adi, "ReVISE: Self-Supervised Speech Resynthesis with Visual Input for Universal and Generalized Speech Enhancement," *arXiv:2212.11377*, Dec. 2022.

[20] A. Pasad, B. Shi, and K. Livescu, "Comparative Layer-Wise Analysis of Self-Supervised Speech Models," in *Proc. of ICASSP*, Rhodes, Greece, June 2023, pp. 1–5.

[21] D. Serdyuk, O. Braga, and O. Siohan, "Audio-Visual Speech Recognition Is Worth 32x32x8 Voxels," in *Proc. of ASRU*, Cartagena, Colombia, Dec. 2021, pp. 796–802.

[22] D. Serdyuk, O. Braga, and O. Siohan, "Transformer-Based Video Front-Ends for Audio-Visual Speech Recognition," *arXiv:2201.10439*, Jan. 2022.

[23] P. Ma, A. Haliassos, A. Fernandez-Lopez, H. Chen, S. Petridis, and M. Pantic, "Auto-AVSR: Audio-Visual Speech Recognition With Automatic Labels," in *Proc. of ICASSP*, Rhodes, Greece, June 2023, pp. 1–5.

[24] A. Zadeh, Y. S. Cao, S. Hessner, P. P. Liang, S. Poria, and L.-P. Morency, "CMU-MOSEAS: A Multimodal Language Dataset for Spanish, Portuguese, German and French," in *Proc. of EMNLP*, virtual, Nov. 2020, vol. 2017, pp. 1801–1812.

[25] M. Anwar, B. Shi, V. Goswami, W.-N. Hsu, J. Pino, and C. Wang, "MuAViC: A Multilingual Audio-Visual Corpus for Robust Speech Recognition and Robust Speech-to-Text Translation," *arXiv:2303.00628*, Mar. 2023.

[26] P. Ma, S. Petridis, and M. Pantic, "Visual Speech Recognition for Multiple Languages in the Wild," *arXiv:2202.13084*, Feb. 2022.

[27] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. De Laroussilhe, A. Gesmundo, M. Attariyan, and S. Gelly, "Parameter-Efficient Transfer Learning for NLP," in *Proc. of ICML*, Long Beach, CA, USA, June 2019, pp. 2790–2799.

[28] X. L. Li and P. Liang, "Prefix-Tuning: Optimizing Continuous Prompts for Generation," in *Proc. of ACL-IJCNLP*, virtual, Aug. 2021, pp. 4582–4597.

[29] C.-L. Fu, Z.-C. Chen, Y.-R. Lee, and H.-y. Lee, "AdapterBias: Parameter-Efficient Token-Dependent Representation Shift for Adapters in NLP Tasks," in *Findings of the ACL-NAACL*, Seattle, WA, USA, July 2022, pp. 2608–2621.

[30] J. He, C. Zhou, X. Ma, T. Berg-Kirkpatrick, and G. Neubig, "Towards a Unified View of Parameter-Efficient Transfer Learning," in *Proc. of ICLR*, virtual, Apr. 2022, pp. 1–15.

[31] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "Wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations," in *Proc. of NeurIPS*, virtual, Dec. 2020, pp. 12449–12460.

[32] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao, et al., "WavLM: Large-Scale Self-Supervised Pre-Training for Full Stack Speech Processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.

[33] H. Le, J. Pino, C. Wang, J. Gu, D. Schwab, and L. Besacier, "Lightweight Adapter Tuning for Multilingual Speech Translation," in *Proc. of ACL-IJCNLP*, virtual, Aug. 2021, pp. 817–824.

[34] Z.-C. Chen, Y.-S. Sung, and H.-Y. Lee, "CHAPTER: Exploiting Convolutional Neural Network Adapters for Self-Supervised Speech Models," *arXiv:2212.01282*, Dec. 2022.

[35] J. Peng, T. Stafylakis, R. Gu, O. Plchot, L. Mošner, L. Burget, and J. Černockỳ, "Parameter-Efficient Transfer Learning of Pre-Trained Transformer Models for Speaker Verification Using Adapters," in *Proc. of ICASSP*, Rhodes, Greece, June 2023, pp. 1–5.

[36] F.-J. Chang, J. Liu, M. Radfar, A. Mouchtaris, M. Omologo, A. Rastrow, and S. Kunzmann, "Context-Aware Transformer Transducer for Speech Recognition," in *Proc. of ASRU*, Cartagena, Colombia, Dec. 2021, pp. 503–510.

[37] K. M. Sathyendra, T. Muniyappa, F.-J. Chang, J. Liu, J. Su, G. P. Strimel, A. Mouchtaris, and S. Kunzmann, "Contextual Adapters for Personalized Speech Recognition in Neural Transducers," in *Proc. of ICASSP*, Singapore, May 2022, pp. 8537–8541.

[38] A. Pasad, J.-C. Chou, and K. Livescu, "Layer-Wise Analysis of a Self-Supervised Speech Representation Model," in *Proc. of ASRU*, Cartagena, Colombia, Dec. 2021, pp. 914–921.

[39] K. Shim, J. Choi, and W. Sung, "Understanding the Role of Self Attention for Efficient Speech Recognition," in *Proc. of ICLR*, virtual, Apr. 2022, pp. 1–19.

[40] C.-H. H. Yang, B. Li, Y. Zhang, N. Chen, R. Prabhavalkar, T. N. Sainath, and T. Strohman, "From English to More Languages: Parameter-Efficient Model Reprogramming for Cross-Lingual Speech Recognition," in *Proc. of ICASSP*, Rhodes, Greece, June 2023, pp. 1–5.

[41] T. Kudo and J. Richardson, "SentencePiece: A Simple and Language Independent Subword Tokenizer and Detokenizer for Neural Text Processing," *arXiv:1808.06226*, Aug. 2018.

[42] D. Snyder, G. Chen, and D. Povey, "MUSAN: A Music, Speech, and Noise Corpus," *arXiv:1510.0848v1*, pp. 1–4, Oct. 2015.