



Language-Universal Phonetic Representation in Multilingual Speech Pretraining for Low-Resource Speech Recognition

Siyuan Feng, Ming Tu, Rui Xia, Chuanzeng Huang, Yuxuan Wang

Speech and Music Intelligence (SAMI), ByteDance

{fengsiyuan.ee, mingtu, rui.xia, huangchuanzeng, wangyuxuan.11}@bytedance.com

Abstract

We improve low-resource ASR by integrating the ideas of multilingual training and self-supervised learning. Concretely, we leverage an International Phonetic Alphabet (IPA) multilingual model to create frame-level pseudo labels for unlabeled speech, and use these pseudo labels to guide hidden-unit BERT (HuBERT) based speech pretraining in a phonetically-informed manner. The experiments on the Multilingual Speech (MLS) Corpus show that the proposed approach consistently outperforms the standard HuBERT on all the target languages. Moreover, on 3 of the 4 languages, comparing to the standard HuBERT, the approach performs better, meanwhile is able to save supervised training data by 1.5k hours (75%) at most. Our approach outperforms most of the state of the arts, with much less pretraining data in terms of hours and language diversity. Compared to XLSR-53 and a retraining based multilingual method, our approach performs better with full and limited finetuning data scenarios.

Index Terms: Low-resource, multilingual training, IPA, phonetic representation, self-supervised learning.

1. Introduction

Recent advancement of end-to-end (E2E) automatic speech recognition (ASR) has been witnessed in high-resource languages e.g. English and Mandarin. One driving force behind is the availability of large amounts of speech data resources. E2E ASR models are data-hungry [1]. Most of the approximately 7000 languages in the world are considered under-resourced [2], in the senses of transcribed and/or untranscribed speech data. Transcribing speech is time-consuming, expensive, and sometimes impossible e.g. for languages with no orthography. This poses a major challenge in developing high-performance ASR systems for low-resource languages.

To tackle the low-resource ASR task, past studies proposed various categories of approaches. one mainstream research line is multilingual training [3, 4], which aims to increase the training data amount and facilitate knowledge sharing among languages. A second research line is crosslingual transfer [5, 6], transferring one or multiple non-target language’s knowledge to the target language. Recently, the application of self-supervised learning (SSL) in the low-resource ASR task has achieved remarkable performances [7, 8, 9, 10, 11]. SSL utilizes supervision information derived from the input data, making it intrinsically suitable for limited-supervision scenarios. In SSL, contrastive learning [8] and masked prediction [12] are two commonly adopted strategies: contrastive learning tries to distinguish a target sample from distractor samples given an anchor representation (e.g. wav2vec 2.0 [8]), while masked prediction tries to output a distribution over a discrete vocabulary given a

masked input utterance (e.g. hidden-unit BERT (HuBERT) [9]). In addition to the aforementioned research lines, data augmentation and self-training, which share the idea of increasing the supervised data amount, were studied [13, 14].

This study addresses the low-resource ASR task by integrating the ideas of multilingual training and SSL. Specifically, we investigate effective and data-efficient pretraining strategies to improve HuBERT [9], encouraging the model to explicitly learn language-universal phonetic representations through multilingual phonetic sharing. Multilingual phonetic sharing is motivated by the analysis [15] which proved that phonetics of different languages are highly shareable, and such sharing benefits multilingual phone recognition. Phonetic sharing can be realized by the language-independent International Phonetic Alphabet (IPA) system [15, 16]. In our proposed approach, we first train a multilingual, hopefully language-universal IPA model which recognizes IPA symbol sequences from speech utterances. This single IPA model captures a broad range of languages’ phonetic information. Next, we leverage the IPA model to create pseudo frame labels for multilingual untranscribed speech data. The IPA pseudo labels are used to conduct HuBERT pretraining instead of K-means based labels [9]. Compared to K-means based pseudo labels, IPA labels encode much richer phonetic knowledge. Guided by the phonetically-informed IPA labels, the pretrained model is capable of capturing a language-universal phonetic representation, a property that benefits downstream ASR finetuning, especially for a language with very limited finetuning data.

Building an IPA model requires supervised data. Our experiments will demonstrate that training of an IPA model is not data-hungry: 100 hours per language suffice. We will show in an ablation study that good IPA pseudo labels can be inferred by an IPA model which is not fully optimized in terms of the phone recognition performance. Furthermore, compared to a standard HuBERT, our proposed approach performs better in ASR tasks, and is able to save up to 1.5k hours (75%) of supervised finetuning data of some target language. In comparison to another multilingual method which is based on IPA model retraining, our approach performs better and is more robust to limited finetuning data. Notably, a recent study [17] shared a similar idea by replacing K-means labels with phoneme alignments derived by a hybrid ASR model in HuBERT pretraining. Unlike IPA, the phoneme system is language-dependent [18]. Our approach is thus not limited to monolingual ASR as was discussed in [17].

2. Proposed approach

2.1. Language-universal IPA model as pseudo labeler

Training a language-universal IPA model requires N languages’ supervised training data (denoted as \mathcal{F}) and phonetic IPA tran-

scripts. The acquisition of IPA transcripts is by conversion from orthographic transcripts leveraging a Grapheme-to-IPA tool such as [19]. The IPA symbols are language-independent, thus the N languages’ training data can be directly merged for multilingual IPA model training. Through training, the IPA model learns phonetic representations that are (quasi) language universal: fundamental speech sounds from different languages that share the same IPA symbol are mapped to the same output neuron of the IPA model. While any ASR architecture could be adopted in principle, we adopt the attention encoder-decoder E2E architecture, based on the finding in [20] – it showed the superiority of the E2E over the hybrid architecture in multilingual phonetic sharing.

The IPA model is trained with joint connectionist temporal classification (CTC) and attention objectives [21]. After training, we rely on the CTC posteriorgram built on top of the IPA model’s encoder to obtain frame-aligned IPA label sequences of input speech utterances: every frame’s label is determined by the output dimension having the largest probability. Given a set of M distinct languages’ untranscribed speech data (denoted as \mathcal{G}), their corresponding IPA pseudo labels are generated by the same IPA model. Notably, there is no assumption on the relation between \mathcal{F} and \mathcal{G} : In the case where there exists a speech sound in \mathcal{G} which is not covered by the pseudo labeler’s IPA inventories, the pseudo labeler will output an IPA symbol that best resembles the sound.

To address the concern of requiring supervised data that might be seemingly less attractive than supervision-free pseudo labeling methods, we will show in our experiments that IPA model training is *not* data-hungry – 100 hours per language suffice.

This approach is applicable to any languages, including those without knowledge of grapheme-to-IPA conversion, because the multilingual IPA model can estimate pseudo labels that best describe speech sounds of any languages.

2.2. Phonetically-informed HuBERT model

The HuBERT model training comprises pretraining followed by finetuning. In the pretraining stage, HuBERT adopts BERT-like [12] masked prediction. Offline K-means based speech frame clustering is carried out multiple rounds, in order to obtain refined K-means labels as the pretraining target. Let X and Z denote a speech utterance and a K-means label sequence respectively, of length T . The masked prediction loss L in a standard HuBERT is:

$$L = \alpha \cdot \sum_{t \in M} \log p_f(z_t | \tilde{X}, t) + (1 - \alpha) \cdot \sum_{t \notin M} \log p_f(z_t | \tilde{X}, t), \quad (1)$$

where M is a set of indices to be masked for X , \tilde{X} is the masked variant of X based on M . The parameter α is the masked prediction weight. The function p_f denotes the prediction process which outputs a distribution over the target Z . The function p_f is parameterized using the HuBERT model’s waveform encoder, Transformer, codeword embedding layer and projection layer, as follows:

$$p_f := \frac{\exp(\text{sim}(\mathbf{A} \cdot \mathbf{o}_t, \mathbf{e}_c) / \tau)}{\sum_{c'=1}^C \exp(\text{sim}(\mathbf{A} \cdot \mathbf{o}_t, \mathbf{e}_{c'}) / \tau)}, \quad (2)$$

where $\{\mathbf{o}_t | t \in 1..T\}$ is the Transformer output (Transformer takes as input from masked output of the waveform encoder), \mathbf{e}_c is the embedding for codeword c , \mathbf{A} is the projection matrix, $\text{sim}(\cdot, \cdot)$ is the cosine similarity, τ is a scalar.

Table 1: *Speech data used in the experiments. Train, Dev and Test rows are presented in the number of hours.*

Language	PL	PT	IT	SP	FR	DU	GE	EN
Train	104	161	247	918	1.1k	1.6k	2.0k	44.7k
Dev	2.1	3.6	5.2	10.0	10.1	12.8	14.3	15.8
Test	2.1	3.7	5.3	10.0	10.1	12.8	14.3	15.6

After pretraining, HuBERT finetuning is conducted using the CTC loss on a target language’s ASR task. Finetuning is applied to the whole model except the waveform encoder.

In this study, we improve the HuBERT pretraining stage, via replacing the K-means clustering labels by IPA labels as the target of masked prediction. Guided by the language-universal and phonetically-rich IPA labels, the pretrained model benefits from multilingual phonetic sharing, which we believe is highly preferred in the subsequent low-resource ASR finetuning. Furthermore, having a well-trained multilingual IPA model, the IPA pseudo labeling process is much faster than the multi-round clustering process [9].

3. Experimental setup

3.1. Databases and evaluation metric

Experiments are carried out using the open-source Multilingual Speech (MLS) corpus [22]. The corpus covers Polish (PL), Portuguese (PT), Italian (IT), Spanish (SP), French (FR), Dutch (DU), German (GE) and English (EN), all derived from read audiobooks. The amount of training data per language varies from 0.1k to 45k hours (see Table 1). As this paper tackles low-resource ASR, while the English training data size is too large to be considered “low-resource”, the English ASR task is excluded from this paper.

We consider two broad types of multilingual training sets for IPA model training and phonetically-informed HuBERT pretraining: (1) **MLS-7**: we merge training sets of PL, PT, IT, SP, FR, DU and GE, summing up to **6.0k** hours. Besides, we select a random training subset of a certain size (at most) from each language, and merge them as limited-supervision versions of MLS-7. We construct **MLS-7-100h**, **MLS-7-250h** and **MLS-7-500h**, where the suffix denotes the data size per language (if the full size of a language’s training set is smaller than the suffix, we apply the full set). (2) **MLS-8-en12kh**: we select a random subset of 12k hours from the EN training set and merge with **MLS-7**, summing up to **18k** hours. We do not take the full EN training data as it would largely damage the non-EN ASR tasks in our experiments.

For HuBERT finetuning, one language’s training set is used at a time. For each language we also prepare a random training subset of 100 hours or 500 hours (at most) for finetuning. The finetuned ASR model is evaluated on a target language’s test set, using word error rate (WER) as the metric. The *dev* data partition is used to monitor the model training procedure.

3.2. Language-universal IPA model setup

The IPA model consists of a Conformer [23] encoder and a Transformer [24] decoder. It is implemented using ESPnet [25], trained with IPA transcripts converted using [19]. Following [15], every IPA symbol, including the modifier (e.g. [ː]), is modeled as a base unit by the IPA model. The encoder and decoder have 18 layers and 2 layers respectively, with 4 attention heads, 768 attention dimensions and 2048 position-wise feed forward (PFF) dimensions. The Conformer encoder con-

tains a 2-layer CNN with a kernel size of 31. Multiple IPA models are trained, each using one of the multilingual datasets as mentioned in Section 3.1. The decoder output dimension of the model, i.e. the IPA inventory size, is 87 (training data excludes EN) or 95 (training data includes EN). The model is trained for 60 epochs with joint CTC/attention objectives and the CTC weight is 0.1, using the Adam optimizer [26], a peak learning rate of 0.001 and 2000 warm-up steps. The final model is obtained by taking average over models of the last 10 epochs.

3.3. Phonetically-informed HuBERT setup

The HuBERT model is implemented by mainly following the setup from [9]. Two model sizes, i.e. ‘BASE’ and ‘LARGE’, are experimented. The models of both sizes contain a 7-layer 512-channel CNN encoder which outputs frames of 16kHz. The BASE-sized HuBERT contains a 12-layer Transformer with 8 attention heads, 768 attention dimensions and 3072 FFN dimensions, and a projection layer of 256 dimensions. For the LARGE-sized HuBERT, the numbers are 24, 16, 1024, 4096 and 768. Regarding masking, 8% of the timesteps are randomly selected as starting positions, and spans of 10 steps are masked. The HuBERT model is pretrained by using one of the multilingual training sets in {MLS-7, MLS-8-en12kh} and corresponding IPA pseudo labels inferred from one of the IPA models discussed in Section 3.2. The pretraining stops after a fixed 400k iterations, using the Adam optimizer [26], a peak learning rate of 5e-4 (BASE) or 1.5e-3 (LARGE), a warm-up step size of 32k. The parameter α is chosen from {1, 0.5, 0.25, 0} to determine the optimal value (see Section 4.1).

HuBERT finetuning is carried out on one of the 7 languages’ training speech and orthographic transcripts (excluding English) at a time. Our initial experiments revealed that the peak learning rate and the maximum iterations largely affect the performance. Unless otherwise stated, we set the two values as 1.6e-4 and 150k. A freeze-step of 10k is set so that in the first 10k iterations, parameters of only the newly added softmax output layer are updated. The final model selection after finetuning is based on the WERs on the dev set. A language model is *not* used throughout this paper.

3.4. Comparative methods

3.4.1. Standard HuBERT

We first compare our method with standard HuBERT models. We train standard HuBERT models by strictly following the multi-round K-means label generation procedure stated in [9]. The 1st-round K-means generates 100 clusters on 39-dimensional MFCC features of the *MLS-7-100h* set. The 1st-round K-means labels are used to pretrain a BASE-sized HuBERT model with the *MLS-7* set (denoted as **HB-Std-1**). The 2nd-round K-means generates 500 clusters on features extracted from the 6-th Transformer layer output of **HB-Std-1**. The data for the 2nd-round clustering is a 10% random subset of *MLS-7-100h* due to the memory constraint. The 2nd-round K-means labels are used to pretrain a BASE-sized HuBERT with the *MLS-7* set, and is denoted as **HB-Std-2**. The 3rd-round K-means clustering is similar to the 2nd-round clustering except the speech features are extracted from the 9-th Transformer layer of **HB-Std-2**. The 3rd-round K-means labels are used to pretrain a LARGE-sized HuBERT with *MLS-7*, and is denoted as **HL-Std**. We take **HB-Std-2** and **HL-Std** as two baseline pretrained models, and carry out finetuning and evaluation following the setup in Section 3.3.

Table 2: WER% results w.r.t different values of α . The HuBERT model is BASE-sized, pretrained with *MLS-7*. The IPA model for pseudo labeling was trained by *MLS-7*.

α	PL	PT	IT	SP	FR	DU	GE	Avg.
1.0	19.42	22.98	14.59	9.61	10.02	17.20	9.86	14.81
0.5	12.35	21.61	13.77	7.37	9.07	14.24	9.40	12.54
0.25	14.17	22.86	14.28	8.23	9.61	14.42	9.84	13.34
0	17.46	25.30	15.83	9.97	11.29	16.33	11.28	15.35

3.4.2. IPA model retraining with BPE transcripts

We then compare our method with an IPA model retraining method as proposed in our another paper [27]. We take the multilingual IPA model trained by *MLS-7* as the seed model, keep its encoder meanwhile replacing its decoder with a randomly initialized one, and retrain it with a target language’s training set using byte-pair encoding tokenized transcripts. The training setup follows Section 3.2. We name this model as **nH-MLS-7**, where ‘nH’ means non-HuBERT.

4. Experimental results

4.1. Effect of masked prediction weight

This section shows the influence of the masked prediction weight α in phonetically-informed HuBERT pretraining. The comparison is made by taking the BASE-sized architecture, using *MLS-7* as the pretraining data. The IPA pseudo labels are inferred by the IPA model trained with *MLS-7*. The WER results are listed in Table 2.

The proposed approach performs the best when α is set to 0.5. This indicates in the proposed phonetically-informed pretraining, the HuBERT model benefits from joint masked and unmasked prediction. When $\alpha = 1.0$, i.e. masked prediction only, our approach degrades by an absolute 2.3% compared to $\alpha = 0.5$, which is opposed to the finding in K-means based pretraining [9]. We conclude that while masked prediction is crucial in pretraining irrespective of the use of IPA or K-means pseudo labels, when phonetically-driven labels are leveraged, unmasked prediction also plays an important role. Setting $\alpha = 0$ results in the worst performance, similar to [9]. For the rest of this section, α is fixed as 0.5.

4.2. Training an IPA model is not data-hungry

This section probes the effect of the language-universal IPA model (i.e. the pseudo labeler) in phonetically-informed HuBERT pretraining. Four IPA models are compared, each trained with one of the *MLS-7*, *MLS-7-100h*, *MLS-7-250h* and *MLS-7-500h* sets. Each IPA model creates pseudo labels for *MLS-7* which are then used to pretrain a HuBERT model. WER% results of finetuned HuBERT models are shown in Table 3. When the HuBERT size is BASE, reducing the training data amount for the IPA model from *MLS-7* (6.0k hours) to *MLS-7-100h* (0.7k hours) leads to negligible degradation. When the HuBERT size is LARGE, the absolute WER degradation is 0.58%. Table 3 also shows that the proposed approach, irrespective of using full or part of the data in training an IPA model, consistently outperforms the standard HuBERT models (**HB-Std-2** and **HL-Std**), and this is true for both BASE and LARGE sizes.

To gain a deeper understanding on the quality of IPA pseudo labels, we conduct an ablation study, i.e. phone recognition by the above-mentioned IPA models. The average phone token error rate (PTER) results over the 7 languages’ test sets are reported in Table 4. Taking Tables 4 and 3, reducing IPA

Table 3: WER% of HuBERT models with different IPA pseudo labels and baseline with K-means labels. Models were pre-trained with MLS-7. “*” denotes the language of which the IPA model training amount is less than that claimed (suffix).

Train set for IPA model	PL	PT	IT	SP	FR	DU	GE	Avg.
HuBERT Transformer size: BASE								
MLS-7	12.35	21.61	13.77	7.37	9.07	14.24	9.40	12.54
MLS-7-500h	12.47*	20.90*	13.22*	8.47	9.11	14.06	9.42	12.52
MLS-7-250h	13.52*	20.30*	13.16*	8.21	9.05	14.06	9.44	12.53
MLS-7-100h	11.64	22.25	13.53	7.86	9.11	14.18	9.75	12.62
HB-Std-2 (baseline)	17.91	21.73	16.22	7.67	10.46	15.22	11.74	14.42
HuBERT Transformer size: LARGE								
MLS-7	5.35	13.22	10.00	5.49	6.58	11.47	7.77	8.55
MLS-7-100h	6.38	14.38	10.65	5.51	7.11	11.86	8.02	9.13
HL-Std (baseline)	6.81	16.26	10.64	6.05	7.33	11.57	8.78	9.63

Table 4: Phone token error rate % by different IPA models. The result is averaged over the 7 languages’ test sets.

Train set for IPA model	MLS-7	MLS-7-500h	MLS-7-250h	MLS-7-100h
Avg. PTER%	4.16	4.60	4.83	5.96

model’s training data has a much greater negative impact on phone recognition (1.8% absolutely) than that on phonetically-informed speech pretraining (0.08% BASE/0.58% LARGE). This may be explained that some inaccurate pseudo labels are consistent over frames, which may still provide useful guidance for pretraining. Conclusively, training an IPA model for the proposed approach is not data-hungry.

4.3. Comparison to other methods

The comparison of the proposed approach, recent state of the arts (SotA) and the comparative method *nH-MLS-7* [27] (see Section 3.4) are summarized in Table 5. The other comparative method, i.e. the standard HuBERT, is included in Table 3. The model’s number of parameters is enclosed in brackets. In all the XLSR-53 [28], B0 (15-lang init.) [29] and JUST [30] models, the 8 MLS language’s training data is fully used. In the proposed approach, we find it detrimental by including the full EN training set during pretraining: taking LARGE, the averaged WER is 11.13% (not listed in the table) by merging EN and MLS-7, and is 8.55% by using MLS-7 only. We find it beneficial by including a 12k-hour subset of the EN set (i.e. *MLS-8-en12kh*) on the BASE-sized model pretraining. Although *MLS-8-en12kh* does not improve the LARGE-sized model compared to *MLS-7*, we report results of both sizes in Table 5.

From Table 5, our approach “*MLS-7*” with the LARGE size performs better than XLSR-53 and B0 (15-lang init.) on average, and utilize much less pretraining data in terms of hours and language diversity. Compared to XLSR-53, our approach consumes additional supervision information in pretraining. Looking at “*MLS-7-100h*” in Table 3, with only 100-hour per language’s supervision, the proposed approach starts to outperform XLSR-53. Moreover, our systems do not adopt a LM while XLSR-53 adopts one. As for B0 (15 lang init.), it consumes supervised training data of 359k hours covering 15 languages to pretrain a seed encoder. The JUST model performs better than our approach. JUST uses more training resources and a slightly larger model architecture. Lastly, our approach (LARGE) performs better than *nH-MLS-7*.

4.4. Robustness towards limited finetuning data

WER% results of the proposed approach and comparative methods using limited finetuning (FT) data are summarized in Table 6. Notably, all the proposed systems in Table 6 use *MLS-7* for

Table 5: WER% of SotA works, comparative methods and the proposed approach. The column “Hrs” denotes the amount of pretraining data in hours. The number enclosed in brackets denotes the parameter number.

	Hrs	PL	PT	IT	SP	FR	DU	GE	Avg.
XLSR-53 (300M) [28]	56k	17.2	14.7	10.4	6.3	7.6	10.8	7.0	10.6
B0 (15-lang init.; 370M) [29]	359k	10.9	15.5	10.1	4.7	6.1	11.1	5.0	9.1
JUST (600M) [30]	50k	6.6	8.0	8.2	3.7	5.2	9.5	4.1	6.5
nH-MLS-7 (216M) [27]	6k	14.14	14.84	10.89	5.36	4.97	11.37	6.90	9.78
HuBERT Transformer size: LARGE (317M)									
MLS-7	6k	5.35	13.22	10.00	5.49	6.58	11.47	7.77	8.55
MLS-8-en12kh	18k	10.50	17.25	11.70	6.44	6.22	11.71	7.63	10.21
HuBERT Transformer size: BASE (95M)									
MLS-7	6k	12.35	21.61	13.77	7.37	9.07	14.24	9.40	12.54
MLS-8-en12kh	18k	10.39	22.38	12.42	8.98	8.63	13.79	9.23	12.26

Table 6: WER% of the proposed approach and comparative methods w.r.t. limited finetuning (FT) data. “*” denotes a language of which the full data amount is less than 500 hours. All the proposed models used *MLS-7* for pretraining. The first column indicates the training set for training the IPA model.

	FT data in hrs	PL	PT	IT	SP	FR	DU	GE	Avg.
XLSR-53 [28]	100	18.9	15.7	12.0	7.9	9.8	10.9	7.4	11.8
nH-MLS-7	100	16.41	15.54	12.00	7.97	8.62	15.62	10.45	12.37
HuBERT Transformer size: LARGE									
MLS-7-100h	min{500, full}	6.38*	14.38*	10.65*	5.52	7.29	12.72	8.55	9.36
MLS-7	100	5.35	13.12	10.60	5.92	7.57	12.64	8.54	9.11
HuBERT Transformer size: BASE									
MLS-7-100h	min{500, full}	11.64*	22.25*	13.53*	8.06	9.39	15.22	10.57	12.95
MLS-7	100	11.10	22.79	15.31	9.98	12.04	17.55	12.30	14.44

speech pretraining, and the first column denotes the training set for the IPA model. We have the following observations:

(1) Compared to standard HuBERT models, our phonetically-informed HuBERT is able to perform better meanwhile consuming less supervised data. For instance, on German, the WER of “HL-Std” (in Table 3) is 8.78% by adopting the full FT data (2k hours), whereas the WER of the LARGE-sized “*MLS-7-100h*” is 8.55% by adopting only 500 hours of FT data plus 100 hours of supervised data for IPA model training. The 100-hour set is a subset of the 500-hour FT set, so our approach saves 1.5k hours (i.e. 75%) of German supervised data. such observations are also seen on French, Spanish (LARGE) and German, Dutch, French (BASE).

(2) Compared to XLSR-53 and *nH-MLS-7*, the proposed LARGE-sized *MLS-7* system achieves better performances in the 100-hour FT data setting. Moreover, our system has a smaller WER degradation (absolute 0.56%) when the FT data reduced from full to 100 hours, comparing to XLSR-53 (1.2%) and *nH-MLS-7* (2.6%).

5. Conclusions

This paper proposed a speech pretraining approach to improving the HuBERT model for low-resource ASR. At its core, language-universal phonetic pseudo labels inferred from a multilingual IPA model guide the speech pretraining process. Experiments on the MLS corpus have shown that our approach consistently outperforms the standard HuBERT model over all the target languages, and on both model sizes. On German, Spanish, Dutch and French, our approach performed better than (or similar to) the standard HuBERT, and was able to save supervised data by 1.5k hours at most. Our approach, consuming much less pretraining data than XLSR-53 [28] and B0 (15-lang init.) [29], performed better than both of them in terms of the averaged WER. Our approach also outperformed a model retrained from a multilingual IPA model. Compared to XLSR-53 and a model retrained from a multilingual IPA model, our approach was more robust to limited finetuning data scenarios.

6. References

- [1] J. Li *et al.*, “Recent advances in end-to-end automatic speech recognition,” *APSIPA Transactions on Signal and Information Processing*, vol. 11, no. 1, 2022.
- [2] O. Scharenborg, L. Ondel, S. Palaskar, P. Arthur, F. Ciannella, M. Du, E. Larsen, D. Merckx, R. Riad, L. Wang *et al.*, “Speech technology for unwritten languages,” *IEEE/ACM Trans. ASLP*, vol. 28, pp. 964–975, 2020.
- [3] J.-T. Huang, J. Li, D. Yu, L. Deng, and Y. Gong, “Cross-language knowledge transfer using multilingual deep neural network with shared hidden layers,” in *Proc. ICASSP*, 2013, pp. 7304–7308.
- [4] S. Toshniwal, T. N. Sainath, R. J. Weiss, B. Li, P. Moreno, E. Weinstein, and K. Rao, “Multilingual speech recognition with a single end-to-end model,” in *Proc. ICASSP*, 2018, pp. 4904–4908.
- [5] P. Swietojanski, A. Ghoshal, and S. Renals, “Unsupervised cross-lingual knowledge transfer in DNN-based LVCSR,” in *Proc. SLT*, 2012, pp. 246–251.
- [6] W. Hou, H. Zhu, Y. Wang, J. Wang, T. Qin, R. Xu, and T. Shinozaki, “Exploiting adapters for cross-lingual low-resource speech recognition,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 317–329, 2021.
- [7] A. van den Oord, Y. Li, and O. Vinyals, “Representation learning with contrastive predictive coding,” *CoRR*, vol. abs/1807.03748, 2018.
- [8] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, “wav2vec 2.0: A framework for self-supervised learning of speech representations,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 12 449–12 460, 2020.
- [9] 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 Trans. ASLP*, vol. 29, pp. 3451–3460, 2021.
- [10] Y.-A. Chung, Y. Zhang, W. Han, C.-C. Chiu, J. Qin, R. Pang, and Y. Wu, “W2v-bert: Combining contrastive learning and masked language modeling for self-supervised speech pre-training,” in *Proc. ASRU*, 2021, pp. 244–250.
- [11] A. Baevski, W. Hsu, Q. Xu, A. Babu, J. Gu, and M. Auli, “data2vec: A general framework for self-supervised learning in speech, vision and language,” *CoRR*, vol. abs/2202.03555, 2022. [Online]. Available: <https://arxiv.org/abs/2202.03555>
- [12] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” *arXiv preprint arXiv:1810.04805*, 2018.
- [13] L. Meng, J. Xu, X. Tan, J. Wang, T. Qin, and B. Xu, “Mixspeech: Data augmentation for low-resource automatic speech recognition,” in *Proc. ICASSP*, 2021, pp. 7008–7012.
- [14] Z.-Q. Zhang, Y. Song, M.-H. Wu, X. Fang, and L.-R. Dai, “Xlst: Cross-lingual self-training to learn multilingual representation for low resource speech recognition,” *arXiv preprint arXiv:2103.08207*, 2021.
- [15] P. Želasko, L. Moro-Velázquez, M. Hasegawa-Johnson, O. Scharenborg, and N. Dehak, “That Sounds Familiar: An Analysis of Phonetic Representations Transfer Across Languages,” in *Proc. INTERSPEECH*, 2020, pp. 3705–3709.
- [16] S. Feng, P. Želasko, L. Moro-Velázquez, A. Abavisani, M. Hasegawa-Johnson, O. Scharenborg, and N. Dehak, “How phonotactics affect multilingual and zero-shot asr performance,” in *Proc. ICASSP*, 2021, pp. 7238–7242.
- [17] C. Wang, Y. Wang, Y. Wu, S. Chen, J. Li, S. Liu, and F. Wei, “Supervision-Guided Codebooks for Masked Prediction in Speech Pre-training,” in *Proc. INTERSPEECH*, 2022, pp. 2643–2647.
- [18] Wikipedia contributors, “Phoneme — Wikipedia, the free encyclopedia,” 2022, [Online; accessed 21-September-2022]. [Online]. Available: <https://en.wikipedia.org/w/index.php?title=Phoneme&oldid=1099218041>
- [19] M. Hasegawa-Johnson, L. Rolston, C. Goudeseune, G.-A. Levow, and K. Kirchhoff, “Grapheme-to-phoneme transduction for cross-language ASR,” in *Proc. ICSLSP*. Springer, 2020, pp. 3–19.
- [20] P. Želasko, S. Feng, L. M. Velázquez, A. Abavisani, S. Bhati, O. Scharenborg, M. Hasegawa-Johnson, and N. Dehak, “Discovering phonetic inventories with crosslingual automatic speech recognition,” *Computer Speech & Language*, vol. 74, p. 101358, 2022.
- [21] S. Kim, T. Hori, and S. Watanabe, “Joint CTC-attention based end-to-end speech recognition using multi-task learning,” in *Proc. ICASSP*, 2017, pp. 4835–4839.
- [22] V. Pratap, Q. Xu, A. Sriram, G. Synnaeve, and R. Collobert, “MLS: A large-scale multilingual dataset for speech research,” *arXiv preprint arXiv:2012.03411*, 2020.
- [23] P. Guo, F. Boyer, X. Chang, T. Hayashi, Y. Higuchi, H. Inaguma, N. Kamo, C. Li, D. Garcia-Romero, J. Shi *et al.*, “Recent developments on espnet toolkit boosted by conformer,” in *Proc. ICASSP*, 2021, pp. 5874–5878.
- [24] S. Karita, N. Chen, T. Hayashi, T. Hori, H. Inaguma, Z. Jiang, M. Someki, N. E. Y. Soplin, R. Yamamoto, X. Wang *et al.*, “A comparative study on transformer vs rnn in speech applications,” in *Proc. ASRU*, 2019, pp. 449–456.
- [25] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N.-E. Y. Soplin, J. Heymann, M. Wiesner, N. Chen *et al.*, “Espnet: End-to-end speech processing toolkit,” *Proc. INTERSPEECH*, pp. 2207–2211, 2018.
- [26] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv*, vol. abs/1412.6980, 2014.
- [27] S. Feng, M. Tu, R. Xia, C. Huang, and Y. Wang, “Language-universal phonetic encoder for low-resource speech recognition,” in *Proc. INTERSPEECH (to appear)*, 2023.
- [28] A. Conneau, A. Baevski, R. Collobert, A. Mohamed, and M. Auli, “Unsupervised cross-lingual representation learning for speech recognition,” *arXiv preprint arXiv:2006.13979*, 2020.
- [29] B. Li, R. Pang, T. N. Sainath, A. Gulati, Y. Zhang, J. Qin, P. Haghani, W. R. Huang, M. Ma, and J. Bai, “Scaling end-to-end models for large-scale multilingual asr,” in *Proc. ASRU*, 2021, pp. 1011–1018.
- [30] J. Bai, B. Li, Y. Zhang, A. Bapna, N. Siddhartha, K. C. Sim, and T. N. Sainath, “Joint unsupervised and supervised training for multilingual asr,” in *Proc. ICASSP*, 2022, pp. 6402–6406.