MIXTURES OF LIGHTWEIGHT ARTICULATORY EXPERTS FOR MULTILINGUAL ASR

Masato Mimura, Jaeyoung Lee

NTT, Inc., Japan

Ryo Magoshi, Tatsuya Kawahara

School of Informatics, Kyoto University, Japan

ABSTRACT

Multilingual end-to-end ASR removes the need for language-specific models but often relies on large architectures and many active parameters, and can suffer from negative transfer across distant languages. Language-universal articulatory information could help regularize multilingual ASR training. We hypothesize that articulatory features, being language-universal and simpler than grapheme targets, can be modeled with extremely lightweight components when properly positioned in the network hierarchy. We propose mixtures of lightweight articulatory experts (MoLAE) trained with a novel multilabel articulatory CTC objective. Unlike conventional MoE approaches that increase total parameters, MoLAE maintains the baseline model size while reducing active parameters by assigning experts to articulatory feature classes and sharing knowledge across languages. On CommonVoice, MoLAE consistently improves recognition accuracy for both high- and low-resource languages while keeping the total parameter budget fixed and lowering active parameter usage. These results show that linguistic inductive bias can substitute for computational scale in multilingual ASR.

***Index Terms*—** Multilingual ASR, mixture-of-experts, IPA, articulatory features, Conformer

1. INTRODUCTION

Multilingual end-to-end (e2e) ASR allows a single model with a shared grapheme-based vocabulary to transcribe speech across multiple languages. This approach removes the need for language-specific models and improves performance in low-resource languages by leveraging data from high-resource ones [1, 2]. However, as in models such as Whisper [3] and XLSR [4], achieving strong multilingual performance typically requires very large architectures making them impractical for resource-constrained scenarios such as on-device or streaming ASR. In addition, joint training across linguistically distant languages can introduce negative transfer, degrading recognition accuracy [2, 5].

Recent studies have explored techniques to address these issues. Mixture-of-experts (MoE) models improve representational capacity by activating only a sparse subset of experts, thereby maintaining inference-time efficiency while scaling model expressiveness [6–11]. Incorporating language-universal representations such as the international phonetic alphabet (IPA) and articulatory features in addition to language-dependent grapheme tokens has been shown to mitigate negative transfer between distant languages [2, 5]. For example, IPA-based tokenization helps stabilize multilingual training [1, 5], and phonetic knowledge improves recognition accuracy even in large-scale pretrained models [5, 12]. Recent work further suggests that combining sparse MoE architectures with phonetic features can synergistically enhance multilingual ASR performance [11]. Nevertheless, these approaches still tend to require a significantly large total parameter count.

Table 1. Examples of mappings from IPA to articulatory features.

	consonantal	voice	spread glottis	high	long	tense
/p/	+	-	-	-	-	0
/p^h/	+	-	+	-	-	0
/iː/	-	+	-	+	+	+

In this paper, we introduce an efficient MoE-based speech encoder that preserves the overall parameter count of a dense baseline. This is achieved by substantially reducing the size of each expert while redefining their functional role. Rather than assigning experts the complex task of performing e2e ASR with only grapheme targets, we design them to address a more tractable subproblem of predicting individual articulatory features. Our formulation reduces the task to binary classification of determining whether a specific feature is present at each time step, which requires considerably less model capacity. Remarkably, this design cuts active parameters by $4\times$ while consistently achieving 7-9% relative error rate reductions across languages.

2. PRELIMINARIES

2.1. Sparse mixture-of-experts

The mixture-of-experts (MoE) framework enhances model expressiveness for handling highly heterogeneous data [6]. It consists of multiple subnetworks, each specialized in handling a subset of the complete set of training examples. A number of MoE-based architectures for ASR were proposed and shown to be effective in multidomain or multilingual tasks [9–11, 13].

Most of recent MoE-based models sparsely activate a small subset of subnetworks for each input [7, 8]. This preserves the expressive power of MoE while significantly reducing computational overhead during inference. Expert selection is managed by a router network $\mathcal{G}(\cdot)$, implemented as a linear transformation $\boldsymbol{W}^{gate}$:

$$\boldsymbol{p}(\boldsymbol{x}) = \mathrm{softmax}(\boldsymbol{W}^{gate}(\boldsymbol{x})), \quad \mathcal{G}(\boldsymbol{x}) = \text{top-k}(\boldsymbol{p}(\boldsymbol{x})) \tag{1}$$

where $\boldsymbol{x}$ is an input representation and $\text{top-k}(\cdot)$ is a selection function that outputs the largest k values. The final output of the sparse MoE module is calculated as $\sum_{r=1}^{k} \mathcal{G}_r(\boldsymbol{x}) E_{\text{idx}(r)}(\boldsymbol{x})$, where $\text{idx}(r)$ is the expert index of the r-th largest weight, $\mathcal{G}_r(\boldsymbol{x})$ is the r-th largest weight and $E_{idx(r)}(\boldsymbol{x})$ is its output.

2.2. Articulatory features

Articulatory features describe the physical movements of the speech organs during the production of speech sounds. These features are largely shared across languages, even when the writing systems or phoneme inventories differ [14]. Consequently, they provide a strong inductive bias for speech recognition [5, 15, 16].

Constructing a mapping between IPA symbols and articulatory features manually is highly labor-intensive. To address this, we use Panphon [17], a database that associates IPA symbols with articulatory feature representations. Panphon defines 24 articulatory features for more than 6,000 IPA symbols, thereby eliminating the need for manual embedding of phonetic and articulatory knowledge by human experts. Table 1 presents three examples of IPA-to-articulatory feature mappings generated by Panphon. In these mappings, each feature is encoded in a binary fashion: + and - denote the presence and absence of a feature, respectively, while 0 indicates a "don't care" condition, meaning that the presence or absence of the feature is irrelevant for the given IPA symbol.

Finally, IPA symbol sequences themselves are automatically derived from grapheme sequences using grapheme-to-phoneme (G2P) tools [18–20]. This enables articulatory modeling to scale effectively for large-scale ASR training.

3. PROPOSED METHOD

This section describes a parameter-efficient encoder that combines MoE with knowledge of articulatory phonetic features. Our basic approach, following [11], is to replace the second FFN sublayer in the Conformer block [21] with a sparse MoE. However, in this work, we introduce the MoE only in a few layers near the input, with each expert having extremely small capacity, under the assumption that shallower layers encode more language-universal phonetic information [22] and can therefore be parameterized with reduced capacity when trained with appropriate supervision.

3.1. Mixture of lightweight experts (MoLE)

In a standard sparse MoE setup, the dense FFN layer of dimension d_{FFN} in a Transformer [23] is replaced by a mixture of n parallel FFN experts, each of dimension d_{FFN}/k, where only k are activated. This design preserves the number of inference-time parameters while increasing model expressiveness. However, it expands the total parameter count to $n \cdot d_{FFN}/k$, which is prohibitive for memory and computation-constrained tasks such as ASR.

To address this, we constrain each expert to dimension d_{FFN}/n, keeping the total parameter count equal to the dense baseline while reducing the number of active parameters to $k \cdot d_{FFN}/n$. Directly applying this setup to e2e multilingual ASR, however, risks performance degradation due to the limited capacity of each expert.

3.2. Multilabel articulatory CTC

We propose training each expert as an *articulatory expert* that predicts a specific articulatory feature, to improve ASR performance while reducing the size of each expert. As discussed in Section 2.2, predicting articulatory features can be framed as binary classification, which is simpler and more language-universal than predicting hundreds or thousands of grapheme tokens. Such tasks are well-suited to low-capacity models. Moreover, since phones can be characterized by combinations of articulatory features, integrating multiple articulatory experts enhances the accuracy of final grapheme predictions.

Unlike phoneme or grapheme-based labels, predicting articulatory features requires *multilabel* classification at each time step, as each IPA unit is defined by a set of 24 features. Moreover, because articulatory label sequences differ in length from the speech input, an alignment between feature values and the acoustic signal must be estimated. A straightforward approach is to apply separate CTC losses [24] to the 24 feature sequences. However, this fails to preserve temporal synchronization among the features representing a single IPA unit. Here, we develop an *articulatory CTC* for e2e articulatory modeling based on the multilabel CTC framework [25].

Table 2. Articulatory feature classes.

class	features
major class	[±syllable], [±sonorant], [±consonantal], [±continuant]
laryngeal	[±voice], [±spread glottis], [±constricted glottis]
major place	[±anterior], [±coronal], [±labial], [±distributed]
minor place	[±high], [±low], [±back]
manner	[±nasal], [±lateral], [±delayed release], [±strident]
minor manner	[±round], [±tense], [±long]
suprasegmental	[±high tone], [±high register], [±velaric]

Let $\boldsymbol{z}_t^f$ denote the 2-dimensional logit vector for articulatory feature f at time step t, where $f \in F = \{\text{syl}, \text{son}, \ldots, \text{velaric}\}$. From this, we compute the probability of the reference label $y_i^f \in \mathbb{Y} = \{-, +\}$. The probability of observing y_i^f at step t is given by $\boldsymbol{p}(y_i^f|\boldsymbol{z}_t^f)$, where $\boldsymbol{p}(\cdot|\boldsymbol{z}_t^f) = \mathrm{softmax}(\boldsymbol{z}_t^f)$. The logit vector $\boldsymbol{z}_t^f$ is obtained via a feature-specific linear transformation $\boldsymbol{W}^f$ applied to the encoder output $\boldsymbol{x}_t$ as $\boldsymbol{z}_t^f = \boldsymbol{W}^f(\boldsymbol{x}_t)$. In addition, we define a separate 2-dimensional logit vector $\boldsymbol{z}_t^\phi$ for CTC blank prediction, with its own linear transformation $\boldsymbol{W}^\phi$, vocabulary $y^\phi \in Y^\phi = \{\text{"blank"}, \text{"non-blank"}\}$, and probability distribution $\boldsymbol{p}(\cdot|\boldsymbol{z}_t^\phi) = \mathrm{softmax}(\boldsymbol{z}_t^\phi)$.

To handle the "don't care" condition, we also introduce an extended label vocabulary $\hat{\mathbb{Y}} = \{-, 0, +\}$. Given a reference IPA label sequence $U = [u_1, ..., u_I]$ of length I, Panphon maps it into 24 articulatory label sequences of equal length, $\hat{Y} = \{\hat{Y}^{syl}, ..., \hat{Y}^{velaric}\}$, where $\hat{Y}^f = [\hat{y}_1^f, ..., \hat{y}_I^f]$ and $\hat{y}_i^f \in \hat{\mathbb{Y}}$. The posterior probability $p(U|\boldsymbol{X})$ for U given the encoder output sequence $\boldsymbol{X} = [\boldsymbol{x}_1, ..., \boldsymbol{x}_T]$ is computed following the standard CTC framework [24]. The CTC blank and non-blank probabilities are derived as:

$$\boldsymbol{p}^{IPA}(\phi|\boldsymbol{x}_t) = \boldsymbol{p}(\text{"blank"}|\boldsymbol{z}_t^\phi) \tag{2}$$

$$\boldsymbol{p}^{IPA}(u_i|\boldsymbol{x}_t) = \boldsymbol{p}(\text{"non-blank"}|\boldsymbol{z}_t^\phi) \cdot \prod_{f \in F} \boldsymbol{s}(\hat{y}_i^f|\boldsymbol{z}_t^f) \tag{3}$$

using the quantity $\boldsymbol{s}(\hat{y}_i^f|\boldsymbol{z}_t^f)$ defined as:

$$\boldsymbol{s}(\hat{y}_i^f|\boldsymbol{z}_t^f) = \begin{cases} \sum_{y_i^f \in \{-,+\}} \boldsymbol{p}(y_i^f|\boldsymbol{z}_t^f) = 1.0 & \text{if } \hat{y}_i^f = 0 \\ \boldsymbol{p}(y_i^f|\boldsymbol{z}_t^f) & \text{otherwise,} \end{cases} \tag{4}$$

where we marginalize the distribution for y_t^f when its corresponding reference value in $\hat{\mathbb{Y}}$ is 0 ("don't care"), since the variable may take either of $-$ or $+$. The articulatory reference $\hat{y}_i^f$ for every $f \in \mathbb{F}$ is automatically obtained from the IPA reference u_i using Panphon.

3.3. Mixtures of lightweight articulatory experts (MoLAE)

It is possible to implicitly induce articulatory knowledge in experts by simply training MoLE using the articulatory CTC. However, for

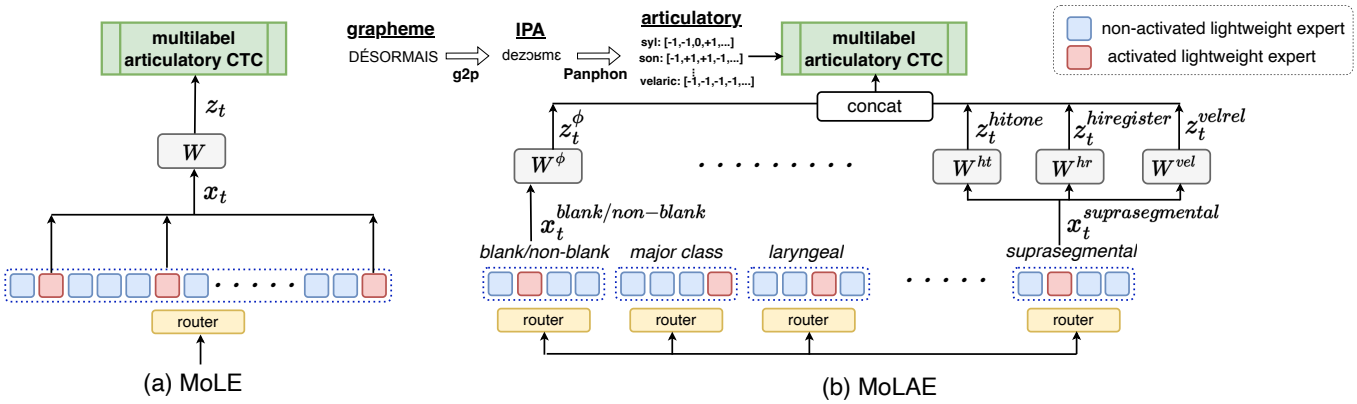

Fig. 1. Structure of the proposed mixtures of articulatory experts.

greater specialization and modularity, we further propose explicitly enforcing each expert to learn a specific feature knowledge.

In theory, this can be achieved using the *target-based routing* algorithm in [11]. Let $\boldsymbol{x}_t^{f,L_a}$ to be the output of the L_a-th layer, where the articulatory loss is applied, computed from the speech input sequence $\boldsymbol{S}$ by activating only the expert assigned to feature f at each MoLE layer. By computing the feature-specific logit as $\boldsymbol{z}_t^f = W^f(\boldsymbol{x}_t^{f,L_a})$, and the non-blank probabilities $\boldsymbol{p}(\hat{y}_i^f|\boldsymbol{z}_t^f)$ in (4) using this logit vector, the loss associated with feature f is back-propagated exclusively to the corresponding expert, ensuring that it acquires knowledge specific to that feature.

However, this approach requires recomputing the encoder outputs up to the L_a-th layer for 25 times with alternately activating experts, making it computationally expensive. To remedy this, we propose grouping the 24 Panphon features into linguistically meaningful categories and assigning a mixture to each group, and reducing the number of encoder recomputations to the number of groups rather than individual features. Among possible groupings, we adopt the classes described in Section 4.1 of [17], namely, *major class*, *laryngeal*, *major place*, *minor place*, *manner*, *minor manner* and *suprasegmental*. The members for each class are listed in Table 2. In addition, we include a special class for blank predictions.

We build a separate mixture for each class $c \in C = \{\text{blank/non-blank}, ..., \text{suprasegmental}\}$, as depicted in Fig. 1. Each mixture has its own trainable router and lightweight experts. We denote the output of the L_a-th layer computed by forwarding only the output of the mixture c and zeroing out all other mixtures as $\boldsymbol{x}_t^{c,L_a}$. The logit vector for feature f is computed as $\boldsymbol{z}_f = W^f(\boldsymbol{x}_t^{cls(f),L_a})$, where $cls(f)$ is a function that returns the class index to which feature f belongs, as shown in Fig. 1.

This approach not only reduces computational cost during training but also facilitates information sharing among related features. Moreover, because experts are assigned to features in a soft, data-dependent manner, the method is expected to be robust to automatically generated and potentially noisy articulatory feature labels.

The entire encoder architecture is depicted in Fig.2. We replace the second FFNs in the first L_a Conformer blocks with MoLAE. For the remaining $L - L_a$ blocks, we stacked the vanilla Conformer blocks. This design allows the Conformer blocks at higher layers to focus capacity on capturing diverse language-specific features, while MoLAE blocks are responsible for capturing more language-universal information from speech. We train the whole network in a multitask training framework. The loss against the primary grapheme target is computed using the output of the final encoder block, while the loss against the articulatory features is applied at the output of the L_a-th block. It is important to note that, for computing the encoder output for grapheme prediction, we use the averaged outputs of all mixtures in MoLAE. The overall loss is given by:

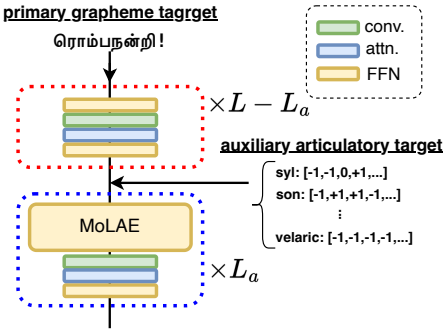

Fig. 2. Encoder architecture.

$$\mathcal{L} = \mathcal{L}_{rnnt}(Y^{grapheme}, \boldsymbol{X}^L) + 0.3 \cdot \mathcal{L}_{ctc}(Y^{grapheme}, \boldsymbol{X}^L) + 0.1 \cdot \mathcal{L}_{ctc}(\hat{Y}^{\text{syl}}, ..., \hat{Y}^{\text{vela}}, \boldsymbol{X}^{cls(\text{syl}),L_a}, ..., \boldsymbol{X}^{cls(\text{vela}),L_a}) \quad (5)$$

4. EXPERIMENTAL EVALUATIONS

4.1. Dataset and models

We conducted experiments on a 15-language dataset derived from CommonVoice v16.1 [26]. To evaluate performance on diverse low-resource languages, we selected 10 languages (*GL-10langs*): Bengali (`bn`), Welsh (`cy`), Finnish (`fi`), Indonesian (`id`), Japanese (`ja`), Polish (`pl`), Russian (`ru`), Swahili (`sw`), Tamil (`ta`), and Thai (`th`). This set represents different language families, scripts, and data sizes. In addition, we included 5 high-resource Western European languages (*WE-5langs*): German (`de`), English (`en`), Spanish (`es`), French (`fr`), and Italian (`it`), to assess model performance on major languages and examine MoLAE's effectiveness in augmenting low-resource training.

Since no single grapheme-to-phoneme (G2P) tool was consistently reliable across languages, we employed three: Charsiu [19] (`cy`, `en`, `es`, `fi`, `fr`, `id`, `it`, `ja`, `pl`, `ru`, `sw`), Phonetisaurus [18] (`bn`, `ta`), and Epitran [20] (`de`, `th`). IPA outputs from these tools were mapped to 24 articulatory features on-the-fly using Panphon [17]. For subword modeling, we constructed vocabularies of 500 word pieces [27] for monolingual models and 5k for multilingual models, except for the Japanese monolingual model, which used 2.6k character tokens.

All models are composed of 12 Conformer blocks, with a model dimension of $d_{model} = 512$, a feed-forward dimension of $d_{FFN} =$

Table 3. Results for 10 low-resource languages (CER (%)). Numbers in brackets indicate training data amounts.

training data	model	# total params	# active params	bn (34h)	cy (11h)	fi (3h)	id (8h)	ja (13h)	pl (29h)	ru (38h)	sw (69h)	ta (81h)	th (37h)	ave.
mono	Conformer	83M	83M	N/A	52.7	N/A	N/A	N/A	18.1	11.2	9.2	14.5	16.1	N/A
multi	Conformer	95M	95M	8.6	12.6	12.4	31.9	40.2	10.9	9.4	7.8	**8.7**	13.0	11.6
	+arti. loss	95M	95M	8.4	12.1	11.8	22.6	39.4	10.3	9.1	8.1	9.1	13.1	11.2
	MoLE	95M	89M	8.7	12.5	14.7	33.0	39.9	11.4	9.4	8.0	**8.7**	13.2	11.8
	+arti. loss	95M	89M	8.2	11.8	11.5	24.0	39.1	**9.8**	8.7	7.9	9.3	12.9	11.1
	MoLAE	95M	89M	**8.1**	**11.7**	**11.1**	**20.9**	**38.6**	9.9	**8.4**	**7.7**	**8.7**	**11.8**	**10.6**

Table 4. Results for 5 Western Europe languages (WER (%)).

data	model	de	en	es	fr	it	ave.
mono	Conformer	12.3	20.0	15.1	17.5	26.3	18.2
multi	Conformer	10.7	19.4	10.7	15.9	12.3	13.8
	+arti. loss	10.2	19.1	10.3	15.6	12.0	13.5
	MoLE	10.6	19.5	10.5	15.8	12.4	13.8
	+arti. loss	10.0	18.7	10.0	15.3	**11.3**	13.1
	MoLAE	**9.8**	**18.6**	**9.9**	**14.9**	**11.3**	**12.9**

2048, and 8 attention heads. We adopt the RNN-T architecture [28], where the prediction network is a one-layer unidirectional LSTM with 512 cells, and the joint network has 640 cells. All models are trained with the Adam optimizer [29], using a linear warmup of 25k steps and a peak learning rate of 0.0015, for a total of 20 epochs.

For MoLE, we set the number of experts to 32, with 8 experts activated at a time. Each expert has a small dimension of $d_{FFN}/32 = 64$. In MoLAE, each of the 8 mixtures contains 4 experts of the same size as those in MoLE, and these experts are implemented to the first 4 Conformer blocks. During inference, feature-specific parameters such as W^f are discarded, and only grapheme sequences are output. This eliminates the need for encoder recomputation for target-based routing at inference time.

4.2. Results for 10 low-resourced languages

Table 3 presents results for 10 low-resourced languages. We report character error rate (CER) rather than word error rate (WER), since some of these languages lack explicit word boundaries in their writing systems. A comparison of monolingual and multilingual baselines shows that multilingual training is highly effective for low-resourced languages [1]. Incorporating articulatory information through multilabel CTC further improves performance for 7 languages, likely due to more efficient training enabled by language-universal features. Comparing the "Conformer" and "MoLE" rows reveals that simply adding MoLE blocks with very compact experts degrades performance, probably because of reduced model capacity. However, when articulatory knowledge is integrated into MoLE, it yields lower CERs than "Conformer + arti. loss" in 8 out of 10 languages, clearly demonstrating a synergistic effect between MoLE and multilabel training on articulatory labels.

Explicitly training experts to capture knowledge of specific articulatory classes ("MoLAE") based on the class-based routing algorithm leads to consistent improvements across all languages, achieving a relative 9% average reduction in CER compared with the multilingual baseline, which is statistically significant at the 1% level.

[1]The monolingual models for bn, fi, id and ja did not even converge.

Table 5. Ablations on MoLE and MoLAE configurations (%).

model	*WE-5langs* (WER)	*GL-10langs* (CER)
MoLE + IPA loss	13.6	11.4
MoLE + arti. loss	13.1	11.1
MoLAE	**12.9**	**10.6**
+ random grouping	13.4	11.2

4.3. Results for 5 Western Europe languages

Table 4 reports results for five Western European languages in terms of WER. The overall tendencies are similar to those observed for *GL-10langs*. However, the benefits of incorporating articulatory knowledge through multilabel CTC, as well as combining MoLE with articulatory loss, are more pronounced in this setting, producing consistent and significant improvements across all five languages. We also observe the clear effectiveness of MoLAE, which achieves a relative 7% reduction in WER compared with the multilingual baseline, which is statistically significant at 1 the % level. It is noteworthy that our method is shown to be effective even for these high-resources languages sharing similar writing systems.

Finally, we conducted ablation studies to examine different training configurations of MoLE and MoLAE. As shown in Table 5, supplying single-label IPA targets to MoLE blocks resulted in smaller improvements compared with using articulatory targets, despite that each set of features was mapped one-to-one from the corresponding IPA symbol with Panphon. This indicates that the MoLE architecture provides a meaningful inductive bias specifically for multilabel training, likely because the trainable router implicitly assigns each label to dedicated experts. For MoLAE, we tested a random grouping strategy in which features were arbitrarily distributed into seven mixtures (three groups of four features and four groups of three). This produced significantly worse results than the proposed linguistically motivated assignment. This shows that clustering features according to linguistically informed criteria is crucial.

5. CONCLUSION

In this paper, we demonstrated that a compact encoder can achieve higher accuracy with fewer active parameters by incorporating articulatory information. Since no established approach for e2e articulatory modeling previously existed, we introduced a novel articulatory CTC loss. This formulation enforces temporal alignment across articulatory features through a shared blank prediction and effectively handles the " don ' t care " label, thereby fully leveraging the supervision provided by Panphon. Building on this, our articulatory experts learn to predict features within specific articulatory classes using a class-based routing algorithm, yielding significant performance gains for low-resource languages from diverse subfamilies and regions, as well as major Western European languages.

6. REFERENCES

[1] Shinji Watanabe, Takaaki Hori, and John R. Hershey, "Language independent end-to-end architecture for joint language identification and speech recognition," in *ASRU*, 2017, pp. 265–271.

[2] Shinji Watanabe Oliver Adams, Matthew Wiesner and David Yarowsky, "Massively Multilingual Adversarial Speech Recognition," in *NAACL*, pp. 96—-108.

[3] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever, "Robust speech recognition via large-scale weak supervision," *arXiv preprint arXiv:2212.04356*, 2022.

[4] Alexis Conneau, Alexei Baevski, Ronan Collobert, Abdelrahman Mohamed, and Michael Auli, "Unsupervised cross-lingual representation learning for speech recognition," in *arXiv preprint arXiv:2006.13979*, 2020.

[5] Jaeyoung Lee, Masato Mimura, and Tatsuya Kawahara, "Leveraging IPA and articulatory features as effective inductive biases for multilingual asr training," in *ICASSP*, 2025.

[6] Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton, "Adaptive mixtures of local experts," in *Neural Computation*, 1991, pp. 79–87.

[7] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean, "Outrageously large neural networks: The sparsely-gated mixture-of-experts layer," in *ICLR*, 2017.

[8] William Fedus, Barret Zoph, and Noam Shazeer, "Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity," in *Journal of Machine Learning Research*, vol. 23, pp. 5232–5270.

[9] Zhao You, Shulin Feng, Dan Su, and Dong Yu, "SpeechMoE: Scaling to large acoustic models with dynamic routing mixture of experts," in *Interspeech*, pp. 2077—-2081.

[10] Zhao You, Shulin Feng, Dan Su, and Dong Yu, "SpeechMoE2: Mixture-of-experts model with improved routing," in *ICASSP*, pp. 7217—-7221.

[11] Masato Mimura, Jaeyoung Lee, and Tatsuya Kawahara, "Switch Conformer with Universal Phonetic Experts for Multilingual ASR," in *Interspeech 2025*, 2025, pp. 1128–1132.

[12] Ryo Magoshi, Shinsuke Sakai, Jaeyoung Lee, and Tatsuya Kawahara, "Multi-lingual and Zero-Shot Speech Recognition by Incorporating Classification of Language-Independent Articulatory Features," in *Interspeech 2025*, 2025, pp. 91–95.

[13] Raphaël Bagat, Irina Illina, and Emmanuel Vincent, "Mixture of LoRA Experts for Low-Resourced Multi-Accent Automatic Speech Recognition," in *Interspeech 2025*, 2025, pp. 1143–1147.

[14] S. Stuker, T. Schultz, F. Metze, and A. Waibel, "Multilingual articulatory features," in *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03).*, 2003, vol. 1, pp. I–I.

[15] M. Ostendorf, "Moving beyound the 'beads-on-a-string model of speech," in *ASRU*, 1999.

[16] Florian Metze and Alex Waibel, "A flexible stream architecture for asr using articulatory featuresh," in *ICSLP*, 2002.

[17] David R. Mortensen, Patrick Littell, Akash Bharadwaj, Kartik Goyal, Chris Dyer, and Lori Levin, "PanPhon: A Resource for Mapping IPA Segments to Articulatory Feature Vectors," in *COLING*, pp. 3475—-3484.

[18] Josef Robert Novak, Nobuaki Minematsu, and Keikichi Hirose, "Phonetisaurus: Exploring grapheme-to-phoneme conversion with joint n-gram models in the wfst framework," in *Natural Language Engineering*, 2016, vol. 22, pp. 907–938.

[19] Jian Zhu1, Cong Zhang, and David Jurgens, "Byt5 model for massively multilingual grapheme-to-phoneme conversion," in *Interspeech*, pp. 446–450.

[20] David R. Mortensen, Siddharth Dalmia, and Patrick Littell, "Epitran: Precision G2P for Many Languages," in *LREC*, 2018.

[21] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang, "Conformer: Convolution-augmented Transformer for Speech Recognition," in *Interspeech*, 2020, pp. 5036–5040.

[22] Kwanghee Choi, Ankita Pasad, Tomohiko Nakamura, Satoru Fukayama, Karen Livescu, and Shinji Watanabe, "Self-Supervised Speech Representations are More Phonetic than Semantic," in *Interspeech 2024*, 2024, pp. 4578–4582.

[23] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, L ukasz Kaiser, and Illia Polosukhin, "Attention is all you need," in *31st Conference on Neural Information Processing Systems (NIPS 2017)*, 2017, vol. 30.

[24] Alex Graves, Santiago Fernandez, Faustino Gomez, and Jurgen Schmidhuber, "Connectionist temporal classification : Labelling unsegmented sequence data with recurrent neural networks," in *Proc. of the 23st International Conference on Machine Learning*, 2006, pp. 369–376.

[25] Curtis Wigington, Brian Price, and Scott Cohen, "Multi-label connectionist temporal classification," in *2019 International Conference on Document Analysis and Recognition (ICDAR)*, 2019, pp. 979–986.

[26] Rosana Ardila, Megan Branson, Kelly Davis, Michael Kohler, Josh Meyer, Michael Henretty, Reuben Morais, Lindsay Saunders, Francis Tyers, and Gregor Weber, "Common Voice: A Massively-Multilingual Speech Corpus," in *LREC*, vol. 23, pp. 4218—-4222.

[27] Rico Sennrich, Barry Haddow, and Alexandra Birch, "Neural machine translation of rare words with subword units," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, 2016, vol. 1, pp. 1715–1725.

[28] Alex Graves, "Sequence transduction with recurrent neural networks," in *ICML*, 2012, pp. 4945–4949.

[29] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.