LaTRO: Language-Token Routing for Scalable and Parameter-Efficient Multilingual ASR

1st Ilyes Oukid
LIPN CNRS UMR 7030, USPN
Villetaneuse, France
oukid@lipn.univ-paris13.fr

2nd Bilal Faye
LIPN CNRS UMR 7030, USPN
Villetaneuse, France
faye@lipn.univ-paris13.fr

3rd Hanane Azzag
LIPN CNRS UMR 7030, USPN
Villetaneuse, France
azzag@lipn.univ-paris13.fr

4th Mustapha Lebbah
DAVID Lab, UVSQ, Paris-Saclay University
Versailles, France
mustapha.lebbah@uvsq.fr

5th Said Yacine Boulahia
Ecole Militaire Polytechnique
Algiers, Algeria
saidyacine.boulahia@emp.mdn.dz

***Abstract*—Multilingual automatic speech recognition (ASR) seeks to transcribe speech from multiple languages within a unified framework, yet existing approaches often suffer from poor scalability and high computational cost. Current state-of-the-art models typically rely on language-specific adapters or complex architectures, which increase the number of parameters as new languages are added and remain ineffective for low-resource and linguistically complex languages, particularly African languages. We propose LaTRO (Language-Token Routing Optimizer), a compact multilingual ASR model that addresses these limitations by (i) maintaining a constant number of trainable parameters regardless of the number of languages, (ii) introducing a learnable language token that guides the model through a shared parameter routing mechanism without language-specific modules, and (iii) supporting both simultaneous multilingual training and progressive language integration. The proposed approach is designed to better handle low-resource and under-explored African languages characterized by rich phonological and morphological variability. Experiments conducted on nine African languages show that LaTRO achieves competitive or superior performance compared to strong state-of-the-art multilingual ASR models, while significantly reducing computational and memory requirements.**

***Index Terms*—multilingual speech recognition, scalable ASR, parameter sharing, compact neural networks**

I. Introduction

Recent advances in automatic speech recognition (ASR) have led to substantial improvements in transcription accuracy; however, these gains are often achieved at the expense of increased model complexity, higher training costs, and growing computational requirements [1]–[4]. As a result, many modern ASR systems are designed as closed solutions, optimized for a fixed set of languages, or remain scalable only through costly architectural expansions. Such designs limit flexibility and hinder the deployment of ASR in open and evolving multilingual environments.

Multilingual ASR has emerged as a promising direction to improve efficiency by sharing representations across languages [5], [6]. Nevertheless, most state-of-the-art multilingual models rely on language-specific adapters or auxiliary modules to encode linguistic information [7]–[9]. While effective in controlled settings, these approaches inherently introduce parameter growth and increased system complexity as new languages are added. This scalability issue is further exacerbated for low-resource languages, where limited data availability and high linguistic variability make model adaptation particularly challenging.

These limitations are especially pronounced for African languages, which are characterized by rich phonological and morphological structures, tonal phenomena, and diverse writing systems. Despite their linguistic complexity and societal importance, African languages remain largely under-explored in multilingual ASR research, and even the most powerful contemporary models often struggle to achieve robust performance on them. Addressing both scalability and low-resource challenges is therefore essential for enabling inclusive and practical multilingual ASR systems.

In this work, we propose **LaTRO** (Language-Token Routing Optimizer), a compact and unified multilingual ASR framework designed to overcome these challenges. LaTRO is built upon three key principles: (i) a low-cost multilingual architecture whose number of trainable parameters remains constant regardless of the number of supported languages; (ii) a learnable language token that conditions the model and acts as a routing mechanism to guide shared parameters toward language-relevant representations, without introducing any language-specific adapters; and (iii) flexible training strategies, including simultaneous multilingual training and progressive language integration, enabling open-ended expansion of the system.

We evaluate LaTRO on nine African languages and compare it against strong state-of-the-art multilingual ASR models. Experimental results demonstrate that the proposed approach achieves competitive or superior recognition performance while significantly reducing computational and memory requirements. These results further highlight its effectiveness and scalability for low-resource multilingual ASR, with a particular focus on typologically diverse African languages.

II. Background

A. Pretrained Multilingual ASR Models

Recent multilingual ASR systems commonly rely on pretrained speech representations learned from large-scale unlabeled audio data. Models such as wav2vec 2.0 [10] have shown strong cross-lingual transfer capabilities by learning shared acoustic features across languages. While these representations provide a powerful initialization, achieving competitive multilingual ASR performance typically requires additional adaptation strategies that incorporate language-specific information.

B. Full Fine-Tuning in Multilingual Settings

A straightforward approach to multilingual ASR consists of fully fine-tuning all model parameters using multilingual or language-specific labeled data [11], [12]. Although effective in static and controlled scenarios, full fine-tuning incurs high computational cost and scales poorly as the number of languages increases. Moreover, this strategy often suffers from negative interference between languages, especially in incremental or sequential training setups, making it unsuitable for open and continuously evolving multilingual environments.

C. Language-Specific and Adapter-Based Methods

To mitigate cross-lingual interference, several approaches introduce language-specific adapters or auxiliary modules on top of shared acoustic encoders. Notably, Massively Multilingual Speech (MMS) [13] employs language-dependent adapters to enable language-aware modeling. While such methods improve performance in multilingual settings, they inherently increase the number of parameters and system complexity as new languages are added. This scalability issue is particularly problematic for low-resource and under-explored languages, such as many African languages, where both data scarcity and linguistic complexity limit the effectiveness of adapter-based designs.

D. Motivation

The limitations of existing approaches highlight the need for a multilingual ASR framework that is both parameter-efficient and scalable, while remaining effective for low-resource and linguistically complex languages. In this work, we address these challenges by proposing a unified multilingual ASR model that leverages shared parameters and language-aware conditioning without relying on language-specific components, enabling efficient and open-ended multilingual expansion.

III. Method

We introduce **LaTRO** (Language-Token Routing Optimizer), a unified multilingual ASR framework designed to (i) maintain a constant number of trainable parameters as the number of languages increases, (ii) incorporate language awareness through a routing mechanism based on learnable language tokens, and (iii) support scalable learning via simultaneous or progressive multilingual training, with a particular focus on under-resourced languages such as African languages. This section details the model formulation, the routing mechanism, and the associated learning algorithms.

A. Model Components

LaTRO consists of four main components:

- **Acoustic Encoder** $E(\cdot)$**:** a pretrained speech encoder (e.g., wav2vec 2.0 [10]) that extracts frame-level features from raw audio. **Frozen**.
- **Shared Projection Layer** $P(\cdot)$**:** maps encoder outputs to a latent space suitable for routing. **Frozen**.
- **Routing Module** $R(\cdot)$**:** lightweight transformer layers that incorporate language information via a learnable token $\boldsymbol{\tau}_k$. **Trainable**.
- **Decoder** $D(\cdot)$**:** maps routed representations to vocabulary logits. **Frozen except for final output projection parameters**.

All trainable parameters are shared across languages except the language tokens, ensuring a minimal and constant parameter budget. This architecture is particularly suitable for low-resource languages, as it allows effective cross-lingual transfer without introducing additional parameters per language.

B. Notation and Problem Setup

Let $\mathcal{K} = \{1, \dots, K\}$ denote the set of languages, including low-resource African languages. For each language $k \in \mathcal{K}$, we are given a labeled speech dataset $\mathcal{S}_k = \{(x_i^k, y_i^k)\}$, where x denotes an input speech signal and y its transcription. LaTRO aims to learn a single model f_Θ that produces transcriptions for all languages using shared parameters, enabling knowledge transfer from higher-resource to lower-resource languages.

Each language k is associated with a learnable language token $\boldsymbol{\tau}_k \in \mathbb{R}^d$. We denote by $\mathcal{L}_{\text{CTC}}(\cdot)$ the Connectionist Temporal Classification loss [14] used for training.

C. Shared Acoustic Representation

LaTRO relies on a pretrained acoustic encoder $E(\cdot)$ to extract frame-level speech representations. To reduce computational cost and stabilize learning in low-resource settings, the encoder parameters are **frozen** throughout training. Given an input signal x, the encoder produces:

$$\mathbf{H} = E(x) \in \mathbb{R}^{T \times d}. \tag{1}$$

A shared projection layer maps these features to a common latent space compatible with subsequent processing. All parameters in this stage are shared across languages, ensuring parameter efficiency and effective cross-lingual transfer critical when some languages have very limited annotated data.

D. Language-Token Routing Mechanism

To encode language information without introducing language-specific modules, LaTRO employs a language-token-based routing mechanism. For each language k, the corresponding token $\boldsymbol{\tau}_k$ is injected into the acoustic sequence via *concatenation*:

$$\tilde{\mathbf{H}} = [\boldsymbol{\tau}_k; \mathbf{H}] \in \mathbb{R}^{(T+1) \times d}. \tag{2}$$

The augmented sequence is then processed by the trainable routing module $R(\cdot)$:

$$\mathbf{U} = R(\tilde{\mathbf{H}}). \tag{3}$$

After processing, the token position is removed:

$$\mathbf{U} \leftarrow \mathbf{U}_{2:T+1}. \tag{4}$$

Here, the language token acts as a *router*, steering the shared parameters toward language-relevant representations. Only $R(\cdot)$ and the language tokens are trained, while the encoder, projection, and decoder remain frozen. This enables robust adaptation even when $|\mathcal{S}_k|$ is small, as for many African languages.

E. Decoder and Output Distribution

The routed representations are passed to the shared decoder followed by a linear projection to the output vocabulary:

$$P(y|x,k) = \mathrm{Softmax}(WD(\mathbf{U}) + b), \tag{5}$$

where $D(\cdot)$ is the decoder and (W, b) are trainable output parameters. The model is optimized with the CTC objective [14]:

$$\mathcal{L}_{\text{CTC}} = \mathcal{L}_{\text{CTC}}(P(y|x,k), y). \tag{6}$$

F. Learning Strategies

LaTRO trains its only learnable components the routing module $R(\cdot)$ and the language tokens $\{\boldsymbol{\tau}_k\}$ using two complementary multilingual strategies that ensure both cross-lingual generalization and scalability to new languages:

1. **Simultaneous multilingual training:** all languages are learned jointly by sampling mini-batches that include at least one example from each language, which prevents high-resource languages from dominating updates. This strategy promotes cross-lingual transfer, especially beneficial for low-resource African languages. See Algorithm 1.

2. **Progressive multilingual training:** languages are introduced sequentially. When a new language is added, the model initializes from previously trained parameters, and a small replay buffer containing examples from previously seen languages is used to prevent catastrophic forgetting. This allows the system to incorporate new low-resource languages incrementally without retraining from scratch. See Algorithm 2.

Both strategies leverage the same shared architecture and routing mechanism, ensuring that the number of trainable parameters remains constant regardless of the number of languages.

G. Inference

During inference, LaTRO produces transcriptions for any language k by selecting the corresponding learnable language token $\boldsymbol{\tau}_k$ and performing a forward pass through the shared architecture. This includes encoding the audio with $E(\cdot)$, projecting with $P(\cdot)$, routing through $R(\cdot)$, and decoding with $D(\cdot)$ to obtain probabilities over the vocabulary:

$$\hat{y} = \arg\max_{y} P(y|x,k). \tag{7}$$

Because LaTRO relies on a fully shared architecture and fixed encoder/decoder parameters, adding new languages such as additional African languages requires only a new language token. No architectural changes or extra parameters are needed, allowing scalable and low-cost deployment in multilingual or evolving low-resource environments.

Algorithm 1 Simultaneous Multilingual Training of LaTRO

Input: Datasets $\{\mathcal{S}_k\}_{k\in\mathcal{K}}$, batch size B
Initialize frozen encoder E and projection P
Initialize trainable routing module R and language tokens $\{\boldsymbol{\tau}_k\}$
while not converged **do**
  Sample mini-batch (x, y, k) by **ensuring all** $k \in \mathcal{K}$ **are represented at least once**
  $\mathbf{H} \leftarrow E(x)$
  $\mathbf{H}_p \leftarrow P(\mathbf{H})$
  $\tilde{\mathbf{H}} \leftarrow [\boldsymbol{\tau}_k; \mathbf{H}_p]$
  $\mathbf{U} \leftarrow R(\tilde{\mathbf{H}})$
  Remove token position: $\mathbf{U} \leftarrow \mathbf{U}_{2:T+1}$
  $\hat{y} \leftarrow D(\mathbf{U})$
  Compute $\mathcal{L}_{\text{CTC}}(\hat{y}, y)$
  Update R and $\boldsymbol{\tau}_k$
end while

Algorithm 2 Progressive Multilingual Training of LaTRO

Input: Ordered languages $\{k_1, \ldots, k_K\}$, datasets $\{\mathcal{S}_k\}$, replay ratio ρ
Initialize frozen encoder E and projection P
Initialize trainable routing module R and empty replay buffer $\mathcal{B} \leftarrow \emptyset$
for $i = 1$ to K **do**
  Initialize language token $\boldsymbol{\tau}_{k_i}$
  Construct training set:
  if $i = 1$ **then**
    $\mathcal{T} \leftarrow \mathcal{S}_{k_1}$
  else
    Sample $\rho|\mathcal{S}_{k_i}|$ examples from $\mathcal{B}$
    $\mathcal{T} \leftarrow \mathcal{S}_{k_i} \cup \text{Sample}(\mathcal{B})$
  end if
  Train LaTRO on $\mathcal{T}$ using Algorithm 1
  Update replay buffer:
    Add $\rho|\mathcal{S}_{k_i}|$ examples from $\mathcal{S}_{k_i}$ to $\mathcal{B}$
end for

IV. Experiments

This section evaluates the proposed multilingual ASR framework with a focus on recognition performance, scalability, and computational efficiency. All experiments are conducted under controlled and identical conditions to ensure fair and reproducible comparisons across different training configurations and baselines.

A. Evaluation Metrics

System performance is primarily evaluated using Word Error Rate (WER), which measures transcription accuracy and is the standard metric for automatic speech recognition. In addition, BLEU [15] and ROUGE [16] scores are reported to assess sentence-level similarity and output fluency. These complementary metrics provide a broader view of recognition quality beyond word-level accuracy.

B. Datasets

Experiments are conducted on a multilingual speech corpus covering nine typologically diverse languages: Igbo [17], Wolof [18], Kabyle [19], Twi [20], Yoruba [21], Hausa [22], Shona [23], Fon [24] and Xhosa [25]. These languages exhibit substantial phonetic and linguistic diversity, making the benchmark suitable for evaluating multilingual scalability and parameter-sharing capabilities. On average, each language provides approximately 15 hours of transcribed speech. All audio signals are resampled to 16 kHz, and consistent training, validation, and test splits are used across languages. A unified character-level vocabulary is constructed by merging character inventories from all languages, resulting in a shared vocabulary of 204 symbols. The same tokenizer is applied across all datasets to ensure consistent text processing and fair multilingual comparison.

C. Experimental Setup

All models are trained under identical settings to ensure fair comparison. We use the AdamW optimizer with a learning rate of 1×10^{-3}, weight decay of 1×10^{-3}, and an effective batch size of 32 via gradient accumulation. Each training run is performed for 100 epochs. LaTRO builds upon a pretrained **wav2vec 2.0** [10] backbone. The acoustic encoder $E(\cdot)$ (feature extractor) and the shared projection layer $P(\cdot)$ are both frozen during training to reduce computational cost and stabilize learning. The decoder $D(\cdot)$ is initialized from wav2vec encoder layers, with only the final linear projection layer trained to produce output logits. Language tokens $\boldsymbol{\tau}_k \in \mathbb{R}^{768}$ are initialized with a Gaussian distribution $\mathcal{N}(0, 0.02)$. These tokens, together with the routing module $R(\cdot)$, are the only trainable components. The routing module consists of four Transformer blocks with eight self-attention heads and hidden dimension 768, shared across all languages.

All experiments are conducted on a single NVIDIA A100 GPU (80GB), ensuring controlled and reproducible evaluation.

D. Training Results

1) Simultaneous Multilingual Training: We first evaluate LaTRO in a **simultaneous multilingual training** setting, where all languages are learned jointly within a single shared model (cf. Algorithm 1). This scenario assesses the model's ability to leverage cross-lingual information while avoiding negative interference between languages in a fully shared parameter space. In addition to MMS, we include Whisper as a strong large-scale multilingual baseline trained under the same conditions. As shown in Table I, LaTRO achieves the lowest average WER across all languages, corresponding to a relative reduction of 9.4% compared to MMS, 19.7% compared to Whisper-small, 7.0% compared to Whisper-medium, and 2.0% compared to Whisper-large, while also obtaining the highest ROUGE scores and a competitive BLEU score. These results indicate that the proposed routing mechanism enables effective multilingual knowledge sharing without sacrificing recognition quality. The per-language WER breakdown in Table II further highlights the robustness of LaTRO, with lower or comparable error rates for the majority of languages when compared to both MMS and Whisper variants. This consistency across typologically diverse and low-resource languages suggests that LaTRO effectively mitigates cross-lingual interference in a fully shared training regime. Overall, these results confirm that LaTRO generalizes well in the simultaneous multilingual setting, achieving strong performance without relying on language-specific modules or parameter growth.

TABLE I
AVERAGE MULTILINGUAL RESULTS UNDER SIMULTANEOUS TRAINING. ROUGE-1, ROUGE-2, AND ROUGE-L ARE ABBREVIATED AS R-1, R-2, AND R-L, RESPECTIVELY.

Model	WER ↓	BLEU ↑	R-1 ↑	R-2 ↑	R-L ↑
Whisper-small	0.504	0.345	0.612	0.440	0.609
Whisper-medium	0.377	0.463	0.706	0.551	0.704
Whisper-large	0.327	**0.517**	0.745	0.606	0.743
MMS	0.401	0.436	0.743	0.587	0.741
LaTRO	**0.307**	0.489	**0.766**	**0.622**	**0.764**

TABLE II
PER-LANGUAGE WORD ERROR RATE (WER ↓) UNDER SIMULTANEOUS MULTILINGUAL TRAINING. WHISPER-SMALL, WHISPER-MEDIUM, AND WHISPER-LARGE ARE ABBREVIATED AS WHISPER-S, WHISPER-M, AND WHISPER-L, RESPECTIVELY.

Lang.	Whisper-s	Whisper-m	Whisper-l	MMS	LaTRO
Igbo	0.743	0.599	0.613	0.586	**0.503**
Wolof	0.539	0.373	0.349	0.405	**0.320**
Kabyle	0.890	0.724	0.561	0.619	**0.503**
Twi	0.543	0.327	0.266	0.345	**0.257**
Yoruba	0.364	0.285	0.272	0.339	**0.245**
Hausa	0.177	0.136	0.113	0.199	**0.075**
Shona	0.446	0.392	**0.305**	0.353	0.321
Fon	0.402	0.239	**0.200**	0.398	0.207
Xhosa	0.432	0.322	**0.262**	0.368	0.329

2) Progressive Multilingual Training: Table III reports results under a **progressive multilingual training** scenario, where languages are introduced incrementally while previously learned languages remain involved through replay (ref. Algorithm 2). This setting reflects realistic deployment conditions in which new languages are added over time without retraining from scratch. In this scenario, we compare LaTRO exclusively with MMS, as it is one of the few multilingual ASR frameworks that supports progressive language integration. However, MMS achieves this capability by introducing language-specific adapter modules, which increases the total number of parameters as new languages are added. In contrast, LaTRO integrates new languages solely through learnable language tokens, while keeping a constant parameter budget. Whisper is not considered in this setting, as it does not support progressive training or incremental language adaptation.

Across all evaluated languages, LaTRO consistently achieves lower WER than MMS while maintaining comparable BLEU and ROUGE scores. Pronounced WER reductions are observed for Kabyle (21.9%), Twi (15.1%), Hausa (8.9%), and Igbo (7.2%). These gains highlight the effectiveness of the proposed routing mechanism in leveraging shared representations when

TABLE III
PROGRESSIVE MULTILINGUAL ASR RESULTS COMPARING MMS AND LATRO ACROSS ALL EVALUATED LANGUAGES.

Lang.	Model	WER ↓	BLEU ↑	R-1 ↑	R-2 ↑	R-L ↑
Igbo	MMS	0.542	0.277	0.669	0.487	0.666
	LaTRO	**0.470**	**0.288**	**0.676**	**0.500**	**0.673**
Wolof	MMS	0.278	**0.586**	**0.848**	**0.740**	**0.848**
	LaTRO	**0.247**	0.554	0.796	0.661	0.795
Kabyle	MMS	0.551	0.277	0.695	0.507	0.694
	LaTRO	**0.332**	**0.394**	**0.769**	**0.609**	**0.768**
Twi	MMS	0.443	0.340	0.697	0.497	0.693
	LaTRO	**0.292**	**0.480**	**0.786**	**0.635**	**0.784**
Yoruba	MMS	0.297	0.567	0.806	0.690	0.805
	LaTRO	**0.260**	**0.583**	**0.801**	**0.684**	**0.799**
Hausa	MMS	0.143	0.809	**0.958**	**0.933**	**0.958**
	LaTRO	**0.054**	**0.897**	0.954	0.920	0.954
Shona	MMS	0.310	**0.520**	**0.760**	**0.608**	**0.759**
	LaTRO	**0.303**	0.511	0.744	0.586	0.742
Fon	MMS	0.197	**0.760**	**0.938**	**0.906**	**0.937**
	LaTRO	**0.170**	0.696	0.881	0.802	0.880
Xhosa	MMS	0.336	**0.468**	**0.725**	**0.548**	**0.724**
	LaTRO	**0.316**	0.445	0.706	0.523	0.706

new languages are introduced. Overall, the results demonstrate that LaTRO remains stable under progressive language addition and does not suffer from catastrophic forgetting, despite the absence of language-specific parameters. This confirms the suitability of the proposed approach for scalable and evolving multilingual ASR, particularly in low-resource settings.

E. Zero-Shot Evaluation on FLEURS

We further evaluate LaTRO on the FLEURS benchmark [26] in a zero-shot cross-corpus setting. This evaluation focuses on Igbo, Wolof, Yoruba, Hausa, Shona, and Xhosa. Although these languages are observed during training, testing is performed on a different corpus, allowing us to assess robustness and generalization under distribution shifts, which is particularly relevant for low-resource African languages.

We compare LaTRO with several strong multilingual ASR systems, including Whisper [11] (small: 244 M, medium: 769 M, large: 1.5 B parameters), MMS [13] (approximately 1 B total parameters, with around 30 M trainable parameters in our experimental setting that increase as new languages are added), Seamless-M4T v2 Large [3] (2.3 B parameters), and Omnilingual ASR [4] in multiple configurations (CTC-1B, CTC-7B, LLM-1B, and LLM-7B).

As reported in Table IV, LaTRO consistently outperforms Whisper variants, MMS, and Seamless-M4T v2 Large across all reported metrics, despite relying on a fixed set of approximately **30 M trainable parameters**. Its performance is comparable to Omnilingual ASR CTC-1B, which slightly surpasses LaTRO while requiring nearly **1 B trainable parameters**. Larger Omnilingual ASR variants (e.g., Omni-LLM-7B) achieve the strongest overall performance, but at the cost of substantially higher computational and memory requirements.

Figure 1 highlights the stark contrast in model size and trainable parameter budgets between LaTRO and existing multilingual ASR systems. Unlike MMS, whose number of trainable parameters grows with the number of supported languages due to language-specific adapters, LaTRO maintains a *constant* and compact parameter footprint regardless of multilingual coverage. This comparison underscores that LaTRO achieves competitive zero-shot performance with orders of magnitude fewer trainable parameters.

Overall, these results demonstrate that LaTRO provides a strong trade-off between recognition accuracy and scalability, enabling effective zero-shot cross-corpus generalization for African languages while remaining computationally efficient and suitable for open and evolving multilingual ASR settings.

TABLE IV
AVERAGE ZERO-SHOT ASR PERFORMANCE ON THE FLEURS BENCHMARK. ROUGE-1, ROUGE-2, AND ROUGE-L ARE ABBREVIATED AS R-1, R-2, AND R-L, RESPECTIVELY. SEAMLESS-M4T RESULTS ARE AVERAGED OVER SUPPORTED LANGUAGES ONLY.

Model	WER ↓	BLEU ↑	R-1 ↑	R-2 ↑	R-L ↑
Whisper-small	0.931	0.009	0.133	0.045	0.130
Whisper-medium	0.908	0.016	0.176	0.073	0.172
Whisper-large	0.891	0.018	0.199	0.091	0.194
MMS	0.483	0.326	0.559	0.387	0.561
Seamless-M4T	0.842	0.075	0.294	0.118	0.284
Omni-CTC-1B	0.405	0.384	0.690	0.518	0.685
Omni-CTC-7B	0.358	0.438	0.730	0.570	0.726
Omni-LLM-1B	0.353	0.462	0.731	0.582	0.728
Omni-LLM-7B	**0.300**	**0.510**	**0.766**	**0.628**	**0.762**
LaTRO	0.420	0.375	0.667	0.488	0.661

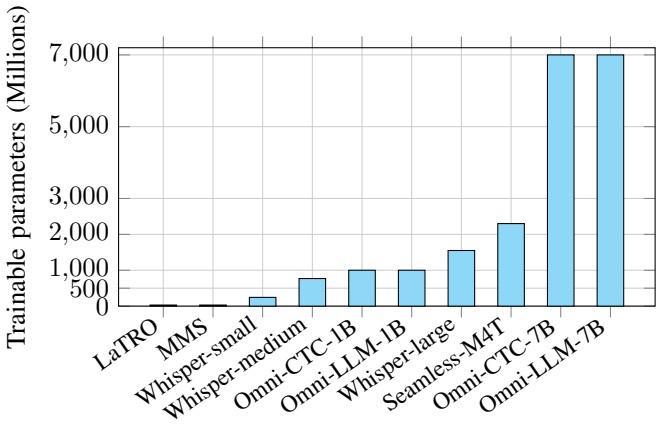

Fig. 1. Comparison of the number of trainable parameters between baseline models and LaTRO (sorted by model size).

F. Ablation Study

To assess the contribution of the proposed language-token routing mechanism, we conduct an ablation study in which the overall LaTRO architecture and training setup are kept unchanged, but the language tokens are removed. In this configuration, the routing module receives only acoustic representations, and no explicit language conditioning is provided

during training or inference. This experiment allows us to isolate the effect of language-aware routing while preserving a fully shared parameterization. Figure 2 reports the resulting Word Error Rates under simultaneous multilingual training. Removing the language tokens consistently leads to higher WER across all evaluated languages, demonstrating a clear degradation in recognition performance. This behavior indicates increased cross-lingual interference when explicit language cues are absent. Overall, these results confirm that language tokens play a critical role in guiding shared representations toward language-relevant subspaces. By enabling language-aware routing without introducing language-specific parameters, the proposed mechanism improves robustness in multilingual and low-resource settings while preserving the simplicity and scalability of the LaTRO architecture.

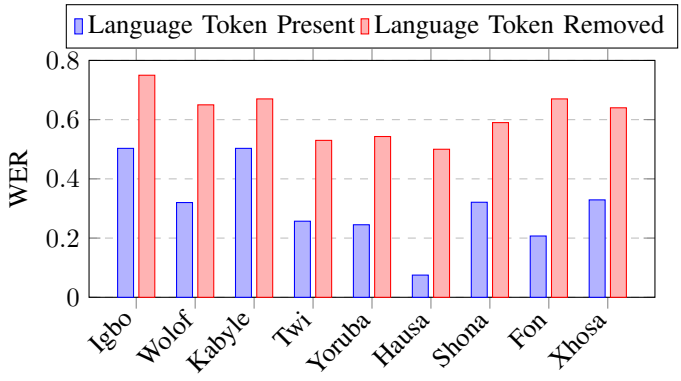

Fig. 2. Ablation study of LaTRO comparing recognition performance with and without language tokens. Lower WER indicates better performance.

V. Conclusion

We present **LaTRO**, a parameter-efficient multilingual ASR framework that relies on a fully shared architecture and a language-token routing mechanism, with a particular focus on low-resource African languages. LaTRO maintains a constant number of trainable parameters while enabling explicit language-aware modeling. Experiments in simultaneous, progressive, and zero-shot cross-corpus settings show that LaTRO achieves competitive or superior recognition performance compared to large multilingual ASR systems, despite using significantly fewer trainable parameters. Improvements are especially pronounced for low-resource languages, confirming the effectiveness of language-token routing in reducing cross-lingual interference. Future work extends LaTRO to a broader set of languages and investigates continuous language integration and robustness under stronger domain shifts.

References

[1] A. Baevski, W.-N. Hsu, A. Conneau, and M. Auli, "Unsupervised speech recognition," *Advances in Neural Information Processing Systems*, vol. 34, pp. 27 826–27 839, 2021.

[2] J. Lin, M. Ge, W. Wang, H. Li, and M. Feng, "Selective hubert: Self-supervised pre-training for target speaker in clean and mixture speech," *IEEE Signal Processing Letters*, 2024.

[3] S. C. Team, "Joint speech and text machine translation for up to 100 languages," *Nature*, vol. 637, no. 8046, pp. 587–593, 2025.

[4] A. Omnilingual, G. Keren, A. Kozhevnikov, Y. Meng, C. Ropers, M. Setzler, S. Wang, I. Adebara, M. Auli, C. Balioglu *et al.*, "Omnilingual asr: Open-source multilingual speech recognition for 1600+ languages," *arXiv preprint arXiv:2511.09690*, 2025.

[5] 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 *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2018, pp. 4904–4908.

[6] A. Tjandra, N. Singhal, D. Zhang, O. Kalinli, A. Mohamed, D. Le, and M. L. Seltzer, "Massively multilingual asr on 70 languages: Tokenization, architecture, and generalization capabilities," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[7] Y. Chen, H. Zhang, X. Yang, W. Zhang, and D. Qu, "Meta-adapter for self-supervised speech models: A solution to low-resource speech recognition challenges," in *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, 2024, pp. 11 215–11 221.

[8] Q. Hu, Y. Zhang, X. Zhang, Z. Han, and X. Liang, "Language fusion via adapters for low-resource speech recognition," *Speech Communication*, vol. 158, p. 103037, 2024.

[9] A. M. Samin, S. Nayak, A. De Marco, and C. Borg, "Investigating adapters for parameter-efficient low-resource automatic speech recognition," in *Proceedings of the 10th Workshop on Representation Learning for NLP (RepL4NLP-2025)*, 2025, pp. 100–107.

[10] 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.

[11] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *International conference on machine learning*. PMLR, 2023, pp. 28 492–28 518.

[12] S. Yusuyin, T. Ma, H. Huang, W. Zhao, and Z. Ou, "Whistle: Data-efficient multilingual and crosslingual speech recognition via weakly phonetic supervision," *IEEE Transactions on Audio, Speech and Language Processing*, 2025.

[13] V. Pratap, A. Tjandra, B. Shi, P. Tomasello, A. Babu, S. Kundu, A. Elkahky, Z. Ni, A. Vyas, M. Fazel-Zarandi *et al.*, "Scaling speech technology to 1,000+ languages," *Journal of Machine Learning Research*, vol. 25, no. 97, pp. 1–52, 2024.

[14] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[15] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2002, pp. 311–318.

[16] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*. Association for Computational Linguistics, 2004, pp. 74–81.

[17] T. Nwankwo, "Igbo tts normalized dataset," https://huggingface.co/datasets/Twelve2five/igbo_tts_normalized, 2025.

[18] Y. Gilbert, "Alffa wolof asr dataset," https://huggingface.co/datasets/yigagilbert/alffa-wolof-asr-dataset-19hr, 2024.

[19] TutlaytAI, "Kabyle asr dataset," https://huggingface.co/datasets/TutlaytAI/kabyle_asr, 2024.

[20] F. Sicoli, "Twi dataset," https://huggingface.co/datasets/fsicoli/twi, 2024.

[21] H. Agili, "Yoruba tts dataset," https://huggingface.co/datasets/Hidi-agili/yoruba_tts_dataset, 2025.

[22] C. Global, "Hausa synthetic asr dataset (xtts)," https://huggingface.co/datasets/CLEAR-Global/Hausa-Synthetic-ASR-Dataset-XTTS, 2025.

[23] R. Speech, "Shona dataset," https://huggingface.co/datasets/realtime-speech/shona1, 2025.

[24] J. Suru, "Fon tts dataset," https://huggingface.co/datasets/jonathansuru/fon_tts, 2024.

[25] W. Mattingly, "Xhosa merged audio dataset," https://huggingface.co/datasets/wjbmattingly/xhosa_merged_audio, 2024.

[26] A. Conneau, M. Ma, S. Khanuja, Y. Zhang, V. Axelrod, S. Dalmia, J. Riesa, C. Rivera, and A. Bapna, "Fleurs: Few-shot learning evaluation of universal representations of speech," in *2022 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2023, pp. 798–805.