Hard to Be Heard: Phoneme-Level ASR Analysis of Phonologically Complex, Low-Resource Endangered Languages

V.S.D.S. Mahesh Akavarapu[1], Michael Daniel[2], Gerhard Jäger[1]
[1]University of Tübingen, [2]University of Jena
mahesh.akavarapu@uni-tuebingen.de, misha.daniel@gmail.com,
gerhard.jaeger@uni-tuebingen.de

Abstract

We present a phoneme-level analysis of automatic speech recognition (ASR) for two low-resourced and phonologically complex East Caucasian languages, Archi and Rutul, based on curated and standardized speech–transcript resources totaling approximately 50 minutes and 1 hour 20 minutes of audio, respectively. Existing recordings and transcriptions are consolidated and processed into a form suitable for ASR training and evaluation. We evaluate several state-of-the-art audio and audio–language models, including wav2vec2, Whisper, and Qwen2-Audio. For wav2vec2, we introduce a language-specific phoneme vocabulary with heuristic output-layer initialization, which yields consistent improvements and achieves performance comparable to or exceeding Whisper in these extremely low-resource settings. Beyond standard word and character error rates, we conduct a detailed phoneme-level error analysis. We find that phoneme recognition accuracy strongly correlates with training frequency, exhibiting a characteristic sigmoid-shaped learning curve. For Archi, this relationship partially breaks for Whisper, pointing to model-specific generalization effects beyond what is predicted by training frequency. Overall, our results indicate that many errors attributed to phonological complexity are better explained by data scarcity. These findings demonstrate the value of phoneme-level evaluation for understanding ASR behavior in low-resource, typologically complex languages.

1 Introduction

Rutul and Archi are two East Caucasian languages with exceptionally rich sound systems that pose major challenges for modern automatic speech recognition (ASR). Archi, in particular, exhibits an unusually complex phonological system with 16 vowel phonemes and—depending on analytical assumptions—between 73 and 81 consonant phonemes, making it one of the largest non-click consonant inventories described to date. Rutul, represented here by its Kina variety, likewise features a large consonant inventory and special articulations such as pharyngealization. Both languages are highly endangered: Archi has only a few thousand speakers and is considered among the most severely endangered languages of Russia (Kibrik et al., 1977; Chumakina et al., 2007), while Rutul is classified as definitely endangered with an estimated ∼30,000 speakers (Alekseeva et al., 2024). Precise counts are difficult due to the geographic dispersion of speaker communities across Daghestan. Crucially, there are no established ASR benchmarks or standardized resources for either language.

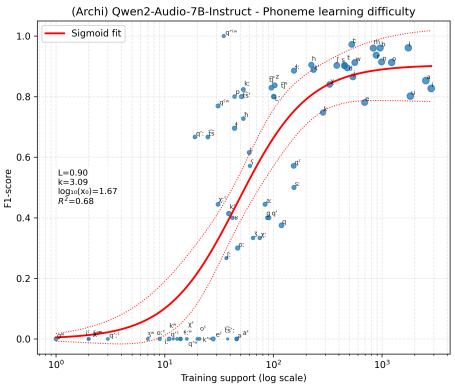

Figure 1: Phoneme-level F1-score as a function of (log) training frequency for one model–language pair, illustrating a characteristic sigmoid-shaped learning trend.

Prior ASR research has largely focused on word- or character-level evaluation and has rarely examined phoneme-level behavior in typologically extreme languages, with only limited exceptions (e.g., Li and Niehues, 2025a). As a result, it remains unclear whether errors attributed to phonological complexity reflect intrinsic difficulty or simply data scarcity.

To address these gaps, we curate and standardize speech–transcript resources for Archi and Kina

Findings of the Association for Computational Linguistics: ACL 2026, pages 3014–3028

Rutul, consolidating material from linguistic documentation projects (Kibrik et al., 2007; Alekseeva et al., 2024) into a form suitable for ASR training and evaluation. The two corpora differ markedly in recording conditions: Kina Rutul consists primarily of spontaneous speech recorded in relatively noisy environments, whereas the Archi data comprise read speech produced by trained speakers under controlled conditions. Despite these differences, we observe broadly comparable trends across the two languages.

We fine-tune and evaluate several state-of-the-art ASR models on these resources, building on recent work showing that useful phonetic recognizers can be obtained from minutes of speech when combined with multilingual pretraining (Boulianne, 2022). By benchmarking multiple architectures under identical conditions, we assess how current ASR systems handle languages with extreme phonological inventories.

Our phoneme-level analysis reveals a consistent pattern across most models and both languages: phoneme recognition accuracy, measured by F1 score, increases sigmoidally with the logarithm of training frequency (Figure 1). Performance is near zero for very rare phonemes, rises sharply once sufficient examples are observed, and saturates for frequent ones. This mirrors frequency effects reported in cognitive and statistical models of language processing, where logistic functions of log-frequency capture performance trends (Heitmeier et al., 2024).

Contributions We summarize our main contributions as follows:

1. We curate speech–transcript resources for Archi and Kina Rutul, enabling systematic ASR benchmarking for two East Caucasian languages previously lacking such resources.

2. We benchmark multiple state-of-the-art ASR architectures including our introduction, a heuristic initialization trick by averaging of CTC based architecture, under extremely low-resource conditions, highlighting the advantages of speech-specialized models.

3. We provide a detailed phoneme-level error analysis, revealing a robust sigmoid-shaped relationship between phoneme recognition accuracy and training frequency.

2 Related work

Approaches to low-resource ASR primarily rely on cross-lingual transfer through multilingual pretraining, often using shared phonemic representations such as IPA or standardized phoneme sets derived from high-resource languages (Siminyu et al., 2021; Li et al., 2022; Taguchi et al., 2023; Pratap et al., 2024). Another common strategy is augmenting end-to-end ASR systems with external n-gram language models to improve decoding in data-scarce settings (Xu et al., 2022; Guillaume et al., 2022; Li and Niehues, 2025a). More recently, heuristic initialization of model layers—particularly output embeddings for unseen phonemes—has been explored to better transfer phonetic knowledge across languages (Yusuyin et al., 2025). Parallel to these developments, several works combine pretrained speech encoders with Large Language Models (LLMs) (Brown et al., 2020; Bai et al., 2023), either keeping the language model frozen (Fong et al., 2025) or adapting it using parameter-efficient techniques such as Low-Rank Adaptation (LoRA) (Hu et al., 2022; Geng et al., 2025).

Despite extensive linguistic documentation, computational resources and ASR studies for East Caucasian languages remain limited. There is no prior ASR work for Archi or Rutul, and only isolated efforts exist for other languages in the family, typically with very small datasets and without phoneme-level evaluation (Li and Niehues, 2025b).

Finally, while ASR performance is sometimes analyzed by broad phonological features/categories (e.g., tone, nasality, vowel length) (Liang and Levow, 2025) or such features are utilized in building ASR systems (Arora et al., 2018), detailed phoneme-level analyses remain rare, especially for phonologically complex and low-resource languages.

3 Datasets

We work with manually transcribed speech data for two endangered East Caucasian languages from the Lezgic branch: Archi and Rutul. Our contribution lies in curating, consolidating, and standardizing existing speech–transcript resources into a form suitable for ASR training and evaluation. The resulting benchmark consists of:

Archi: approximately 45 minutes of training data (545 sentences) and 7 minutes of test data (100 sentences), derived from materials documented in

Lang.	Split	Size	Vocabulary Sizes		
			Words	Phonemes	Composites
Archi	Train	545 / 45m	1445	85	50 (59%)
	Test	100 / 7m	394	70	37 (53%)
Rutul	Train	1388 / 75m	4866	78	40 (51%)
	Test	90 / 7m	441	58	21 (36%)

Table 1: Dataset statistics — split-wise number of sentences, total length in minutes (m), number of unique words, phones and composite (complex) phones.

Kibrik et al. (2007). The recordings contain speech which was read out under controlled conditions.

Kina Rutul: approximately 75 minutes of training data (1,388 sentences) and 7 minutes of test data (90 sentences), based on documentation work reported in Alekseeva et al. (2024). The recordings contains spotaneous speech often recorded in noisy environments.

Detailed split-wise statistics are provided in Table 1. Composite phonemes are those with a diacritic such as ˤ, ʷ, ’ or ː. All possible phonemes per language are listed in the Appendix A.

The original annotations differ substantially across the two corpora. Kina Rutul recordings are aligned with sentence-level annotations in `Praat` TextGrid format, while Archi recordings are aligned using `ELAN`. In both cases, the speech segments are segmented at the sentence level. The transcriptions themselves are heterogeneous, combining IPA symbols, romanized conventions (e.g. š for ʃ), and occasional Cyrillic-based notation (e.g. `|` to mark pharyngealization i.e., ˤ). As part of the curation process, we normalize these annotations into consistent, sentence-level IPA transcriptions paired with the corresponding audio. We adopt IPA to facilitate transfer from multilingual pretrained ASR models.

In terms of speaker coverage, the Kina Rutul data include recordings from approximately 15 adult speakers (with a slight predominance of female speakers), while the Archi data consist of read speech produced by two trained female speakers. Although the corpora are limited in size, their curation enables controlled phoneme-level ASR experiments that were previously not feasible.

4 Models

We evaluated the following model families and their modifications:

wav2vec2-large-ipa (Taguchi et al., 2023) This is a wav2vec 2.0 model (Baevski et al., 2020) fine-tuned for ASR on connectionist temporal classification (CTC) (Graves et al., 2006) using IPA transcriptions from multiple languages, and is therefore suitable for zero-shot evaluation on unseen languages (**-zs** suffixed).

w2v2l-custom For wav2vec2-large-ipa, we additionally define a language-specific phoneme vocabulary derived from the IPA transcriptions. Composite phonemes (e.g., labialized or pharyngealized consonants) are mapped to a reduced vocabulary reflecting the actual phonemic contrasts of each language. For example, the phoneme kʷ is tokenized by the wav2vec2-large-ipa tokenizer as the sequence ‘k’,‘ʷ’, whereas in this model it is represented as a single token. We experiment with two initialization strategies for the output layer of wav2vec2, denoted by suffixes; without any suffix, the final layer is randomly initialized.

w2v2l-custom-avg Columns corresponding to composite phonemes are initialized by averaging the pretrained parameters of their component IPA symbols. Let the weights and biases of the output layer be $W \in \mathbb{R}^{d\times|V|}$ and $b \in \mathbb{R}^{|V|}$, where d is the hidden dimension and V is the reduced, language-specific vocabulary. Let the pretrained weights, biases, and vocabulary be denoted by W^{old}, b^{old}, and V^{old}, respectively. For a phoneme indexed by i in V^{new} that is composed of symbols indexed by $i_1, \ldots, i_k$ in V^{old}, the parameters are initialized as:

$$W_{*i} = \frac{1}{k}\sum_{j=1}^{k} W^{\text{old}}_{*i_j} \quad ; \quad b_i = \frac{1}{k}\sum_{j=1}^{k} b^{\text{old}}_{i_j}.$$

Averaging is our novel step. We also perform zero-shot evaluation with this model enabled by non-random initialization, with suffix **-zs** in the name.

w2v2l-custom-cpy1 In this variant, the output-layer parameters corresponding to the base phoneme (i.e., without diacritics) are copied directly, rather than averaged, following a strategy similar to Yusuyin et al. (2025).

w2v2l-custom-avg-lm Inspired from Xu et al. (2022); Guillaume et al. (2022); Li and Niehues (2025a), we incorporate a word-level n-gram language model (LM) with $n = 3$ on top of w2v2l-custom-avg to reduce word error rate. This differs from previous works slightly as the latter incorporate character/phoneme n-gram, which we did not find improving performance. The CTC output sequence includes a word-separator token ‘|’, which

deterministically segments the phoneme sequence X into words $w_1(X), \ldots, w_{m(X)}(X)$. The vocabulary of LM consists of words occurring in the training transcripts. The mapping from segmented phoneme sequences to LM tokens is handled directly in a dictionary-like manner. During decoding, the following objective is maximized over a phoneme sequence $X = x_1, \ldots, x_l$ and its corresponding word sequence $w_1(X), \ldots, w_{m(X)}(X)$:

$$\max_{X=x_1,\ldots,x_l} \sum_{i=1}^{l} \log p_{\text{ctc}}(x_i) + \beta \cdot m(X) + \alpha \sum_{i=n+1}^{m(X)} \log p_{\text{lm}}(w_i(X) \mid w_{i-1}(X), \ldots, w_{i-n}(X))$$

where p_{ctc} denotes the CTC softmax probabilities produced by wav2vec2, p_{lm} is the probability from the n-gram language model, and α and β are tunable hyperparameters. The optimal sequence is approximated using beam search. We use KenLM (Heafield, 2011) for n-gram language modeling.

whisper-large-v3 (Radford et al., 2023) This is a multilingual pretrained encoder–decoder model that supports IPA output. Due to its inherent subword tokenizer (as well as those of the models described below), vocabulary reduction and output-layer reinitialization are not straightforward; therefore, the model is fine-tuned without modifying its pretrained decoder vocabulary.

Qwen2-Audio-7B-Instruct (Chu et al., 2024) combines an audio encoder (initialized from whisper-large) with a large language model (LLM) (Qwen2; Team et al. (2024)), aligning the encoder outputs with the input representation space of the LLM.

Qwen2.5-Omni-7B (Xu et al., 2025) This model supports text, vision, and audio inputs and can generate spoken responses. In this work, we restrict the model to audio input and text output. For both Qwen-based models, the audio encoder is fine-tuned, while the LLM is fine-tuned using LoRA.

gpt-4o-transcribe (Hurst et al., 2024) This model is accessed via the OpenAI API and is therefore used without fine-tuning. The model predominantly outputs Cyrillic script. Thus, we explicitly prompt it to transcribe in Cyrillic.

Since the conversion between Cyrillic, the official script for these languages (only introduced recently), and IPA is deterministic (see Appendix A for mappings), we additionally fine-tune generative models to output Cyrillic. These variants are denoted as **whisper-large-v3-cyrl**, **Qwen2-Audio-7B-Instruct-cyrl**, and **Qwen2.5-Omni-7B-cyrl**. Evaluations are carried out after converting to IPA.

5 Experiments

5.1 Implementation Details

We create a validation set by holding out 5% of the training sentences. The vocabulary sizes of w2v2l-custom* models are 90 and 84 respectively for Archi and Rutul including special characters. Learning rates are set to 3×10^{-5} for CTC-based models and 5×10^{-6} for Whisper and LLM-coupled audio encoders. All models are optimized using Adam (Kingma, 2014) with a weight decay of 0.01 (Loshchilov and Hutter, 2017). For Qwen-based models, the LoRA parameters are rank $r = 16$, scaling factor $\alpha = 32$, and dropout 0.05 (similar to Geng et al. (2025)). LoRA is applied to all linear layers and optimized using Adam with a learning rate of 1×10^{-4} and no weight decay. The number of trainable and total parameters per model are listed in Table 2. CTC-based models are fine-tuned for 30 epochs, Whisper for 10 epochs, and Qwen-based models for 6 epochs. All models use an effective batch size of 16 via gradient accumulation. CTC-based models are trained on two NVIDIA RTX 2080 GPUs (11 GB $\times$2), while larger models are trained on a single NVIDIA H100 (80 GB).

For the LLM-based models, we use the following prompt, refined through a small number of manual trials with gpt-4o-transcribe:

"`Transcribe the audio in <lang> (a Northeast Caucasian language) into <IPA (International Phonetic Alphabet) | Cyrillic>. Do not translate, interpret, or add punctuation. Output only the phonetic transcription.`"

The 3-gram language model coupled with CTC uses $\alpha = \beta = 0.3$, tuned on the validation sets with a beam size of 10 (restricted to this value for efficiency). The code[1] and the datasets[2] are publicly available.

[1] https://github.com/mahesh-ak/north_caucasian_asr

[2] https://huggingface.co/datasets/mahesh27/archi_rutul_asr

5.2 Evaluation metrics

We evaluate on standard metrics for ASR — word, character and phoneme error rates, respectively WER, CER and PER which are normalized edit distances respectively at levels of words, characters and phonemes. We further store the number of edits — insertions (I), deletions (D) and substitutions (S) — along with true positives (N) for each phoneme to compute phoneme-level precision (pr), recall (re) and F1 scores:

$$\text{pr} = \frac{N}{N+S+I}\ ;\ \text{re} = \frac{N}{N+S+D}; \text{F1} = \frac{2 \cdot \text{pr} \cdot \text{re}}{\text{pr}+\text{re}}$$

5.3 Modeling of Phoneme Frequency Effects

To analyze the relationship between phoneme recognition performance and data availability, we model phoneme-level F1 scores as a function of log training frequency using a logistic function:

$$f(x) = \frac{L}{1 + \exp(-k(x - x_0))}$$

where $x = \log_{10}(\text{training frequency})$, L denotes the asymptotic F1 score, k controls the slope, and x_0 is the midpoint of the transition. We emphasize that the logistic form is used as a descriptive parametric summary of the observed nonlinear trend, rather than as a theoretical assumption. Alternative shapes (e.g., piecewise-linear or threshold-like behavior) may also fit the data; our goal is to capture a consistent frequency–accuracy scaling pattern and estimate interpretable midpoints, rather than to commit to a specific functional form.

Parameters are estimated via non-linear least squares using the Levenberg-Marquardt algorithm (Marquardt, 1963). Model fit is quantified using the coefficient of determination (R^2) between observed and predicted F1 scores. Uncertainty in the fitted curve is assessed using approximate 95% confidence intervals derived via the Delta method (Van der Vaart, 2000).

6 Results

6.1 Overall Performance

Overall results are summarized in Table 2, with statistical significance assessed using paired wilcoxon signed-rank tests (Appendix B). As expected, the zero-shot models perform poorly, yielding near-random WER for both languages, underscoring the extreme low-resource and phonologically complex nature of the tasks.

On **Archi**, whisper-large-v3 achieves the best overall performance. Nevertheless, the w2v2l-custom variants substantially improve over the base wav2vec2-large-ipa model: w2v2l-custom-avg (ours) reduce WER by more than 8 absolute points, with statistically significant gains in both WER and CER ($p < 0.05$) (w2v2l-custom-cpy1 doesn't always give significant gain). Adding a word-level 3-gram language model (w2v2l-custom-avg-lm) yields only marginal additional improvements, consistent with the limited amount of training data. The best (lowest) WER is comparable to results reported for similarly low-resourced languages (e.g., Li and Niehues (2025a)).

On **Rutul**, similar trends hold. The w2v2l-custom-avg model achieves the lowest CER and PER, while w2v2l-custom-avg-lm yields the best WER (0.697), with statistically significant improvements over wav2vec2-large-ipa. In contrast to Archi, whisper-large-v3 performs noticeably worse on Rutul, suggesting weaker transfer. This asymmetry is consistent with later phoneme-level analyses, where Whisper exhibits fewer deviations from frequency-driven learning on Rutul. We further analyze data quality–quantity trade-offs for Rutul in Appendix C.

Across both languages, large audio–language models (Qwen2-Audio and Qwen2.5-Omni) underperform relative to CTC-based models, even after fine-tuning. Their Cyrillic-output variants (-cyrl), corresponding to the recently standardized official script (see §3), generally lag behind their IPA-output counterparts.

Overall performance improves with increasing specialization toward speech, from multimodal Qwen2.5-Omni to audio-centric Qwen2-Audio and dedicated ASR models such as wav2vec2 and Whisper, where simple, linguistically informed adaptations of CTC-based models—language-specific phoneme vocabularies with heuristic initialization—can match or outperform substantially larger pretrained systems. The subsequent phoneme-level analysis clarifies how training frequency and pretraining jointly shape recognition performance.

6.2 Phoneme-level Analysis

Tables 3 and 4 report category-wise F1 scores along with their correlation (Pearson's r) with phoneme complexity. We define phoneme complexity as the number of additional articulatory features (e.g., $^{\text{ʕ}}$, $^{\text{w}}$, ', ː) attached to a base segment, as indicated by

			Archi			Rutul		
Model	**Params.**	**Tunable**	**WER**	**CER**	**PER**	**WER**	**CER**	**PER**
wav2vec2-large-ipa-zs	0.3B	-	1.000	0.593	0.606	1.000	0.656	0.660
wav2vec2-large-ipa	0.3B	0.3B	0.559	0.128	0.135	0.795	0.223	0.220
w2v2l-custom-avg-zs	0.3B	-	1.000	0.544	0.558	1.000	0.563	0.571
w2v2l-custom	0.3B	0.3B	0.593	0.138	0.147	0.780	0.224	0.222
w2v2l-custom-cpy1	0.3B	0.3B	0.462	0.116	0.123	0.738	0.205	0.203
w2v2l-custom-avg (ours)	0.3B	0.3B	0.479	0.116	0.122	0.725	**0.198**	**0.195**
w2v2l-custom-avg-lm (ours)	0.3B	0.3B	0.465	0.116	0.122	**0.697**	0.206	0.206
whisper-large-v3	1.5B	1.5B	**0.402**	**0.099**	**0.107**	0.778	0.253	0.251
whisper-large-v3-cyrl	1.5B	1.5B	0.422	0.111	0.119	0.792	0.232	0.235
Qwen2-Audio-7B-Instruct	8.4B	0.7B	0.579	0.163	0.180	0.778	0.242	0.239
Qwen2-Audio-7B-Instruct-cyrl	8.4B	0.7B	0.539	0.156	0.166	0.828	0.272	0.274
Qwen2.5-Omni-7B	10.8B	0.7B	0.705	0.184	0.199	0.852	0.263	0.257
Qwen2.5-Omni-7B-cyrl	10.8B	0.7B	0.904	0.295	0.291	0.904	0.295	0.291
gpt-4o-transcribe	-	-	0.982	0.435	0.436	0.994	0.519	0.514

Table 2: ASR Performance of models on Archi and Kina Rutul in terms of Word- (WER), Character- (CER) and Phoneme- (PER) error rates. Lower the error rates better the model. Best performances are in **bold**.

Complexity→	**1**	**2**	**2**	**3**	**3**	**4**	**3**	**4**	**2**	**3**	**3**	**2**	**3**	**1**	**2**	**3**	**2**	
Model↓	**C**	**C^{w}**	**C'**	**C'^{w}**	**C':**	**C':^{ˤ}**	**C'^{ˤ}**	**C'^{ˤw}**	**C:**	**C:^{w}**	**C:^{ˤ}**	**C^{ˤ}**	**C^{ˤw}**	**V**	**V:**	**V:^{ˤ}**	**V^{ˤ}**	r
gpt-4o-transcribe	0.449	0.4	0.035	0.0	0.0	0.0	0.0	0.0	0.355	0.0	0.0	0.0	0.0	0.525	0.234	0.0	0.0	-0.76
whisper-large-v3	**0.894**	0.667	**0.823**	0.333	0.5	0.0	0.0	0.667	0.803	0.722	0.769	0.673	0.545	0.736	0.577	**1.0**	0.317	-0.44
whisper-large-v3-cyrl	0.802	**0.863**	0.78	**0.667**	1.0	0.0	0.0	**1.0**	0.817	0.889	**0.833**	0.419	0.667	0.742	**0.583**	0.889	**0.442**	-0.15
Qwen2-Audio-7B-Instruct	0.764	0.0	0.684	0.0	0.333	0.0	0.0	**1.0**	0.63	0.0	0.444	0.286	0.769	0.696	0.248	0.0	0.0	-0.21
Qwen2-Audio-7B-Instruct-cyrl	0.735	0.636	0.615	0.333	**0.625**	0.0	0.222	**1.0**	0.583	0.333	0.4	0.35	0.8	0.709	0.484	0.667	0.156	-0.18
wav2vec2-large-ipa	0.878	0.667	0.846	0.333	0.4	0.0	**0.333**	0.8	**0.862**	**0.933**	0.417	**0.87**	0.933	0.765	0.483	0.857	0.345	-0.38
wav2vec2-large-ipa-zs	0.308	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.188	0.0	0.0	0.0	0.0	0.47	0.153	0.0	0.0	-0.71
Qwen2.5-Omni-7B	0.728	0.0	0.53	0.0	0.222	**0.667**	0.0	0.0	0.632	0.0	0.0	0.348	0.4	0.681	0.366	0.0	0.0	-0.46
Qwen2.5-Omni-7B-cyrl	0.697	0.303	0.368	0.0	0.0	0.0	0.0	0.5	0.489	0.167	0.0	0.283	0.364	0.661	0.412	0.0	0.0	-0.6
w2v2l-custom-avg	0.867	0.222	0.778	0.333	0.417	0.0	0.0	0.857	0.753	0.0	0.462	0.426	0.933	**0.773**	0.389	0.0	0.192	-0.36
w2v2l-custom-avg-zs	0.34	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.489	0.0	0.0	0.0	-0.65
w2v2l-custom	0.835	0.0	0.79	0.25	0.143	0.0	0.0	**1.0**	0.749	0.0	0.455	0.417	0.933	0.71	0.341	0.0	0.0	-0.24
w2v2l-custom-avg-lm	0.88	0.333	0.809	0.333	0.364	0.0	0.0	0.857	0.752	0.0	0.462	0.426	0.933	**0.773**	0.413	0.0	0.192	-0.39
w2v2l-custom-cpy1	0.878	0.333	0.791	0.333	0.417	**0.667**	0.0	0.857	0.841	0.222	0.5	0.447	**1.0**	0.771	0.481	**1.0**	0.354	-0.09

Table 3: Phoneme category-wise averaged F1 scores and their Pearson's correlation coefficients r with complexities (length) - Archi

IPA diacritics, plus 1 for the base segment. In IPA, using a diacritic usually indicates an additional articulation - such as ^{w} for labialization or a feature that distinguishes the phoneme from its more "typologically common" counterpart, as ' for non-pulmonic (ejective) consonant. Such additional features are a reasonable proxy for articulatory complexity as well as for typological markedness. Across both languages, simpler segments are consistently recognized better than marked categories including secondary articulations, length contrasts, and pharyngealized or labialized segments (also see the phonemes with least F1 in Appendix D).

For Rutul, we observe a clear negative correlation between complexity and performance across models, indicating systematic degradation as grapheme length or articulatory complexity increases. In contrast, Archi shows only weak to moderate negative correlations, suggesting that performance degradation is less monotonic and more category-specific rather than strictly driven by length. This weaker trend motivates an alternative treatment and accordingly, in the following section we model recognition performance as a function of log training frequency using a sigmoid formulation, which is independent of phoneme complexity.

In both overall performance (Table 2) and at the phoneme level, zero-shot models—gpt-4o-transcribe, wav2vec2-large-ipa-zs, and w2v2l-custom-avg-zs—serve as baselines, illustrating that without language-specific pre-training it is nearly impossible to reliably recognize articulatorily complex phonemes.

6.3 Training Frequency–F1 Score Analysis

A central observation of this work is the strong relationship between phoneme-level F1 scores and the logarithm of training frequency across models and both languages (Figure 2). As formalized in §5.3, we fit a logistic curve to this relationship. For most fine-tuned models, phoneme recognition accuracy follows a characteristic sigmoid shape: very rare phonemes exhibit near-zero F1, followed by a sharp transition as training support increases, and eventual saturation for frequent phonemes. The resulting fits explain a substantial fraction of phoneme-

Complexity→	1	2	3	2	2	3	3	1	2	2	
Model↓	**C**	**C^{j}**	**C^{j}ː**	**C^{w}**	**C'**	**C'^{j}**	**C'^{w}**	**V**	**Vː**	**V^{ˤ}**	r
gpt-4o-transcribe	0.396	0.0	0.0	0.1	0.0	0.0	0.0	0.457	0.034	0.0	-0.81
whisper-large-v3	0.76	0.429	0.0	0.31	0.37	0.0	0.0	0.715	0.056	0.165	-0.92
whisper-large-v3-cyrl	0.721	0.333	0.0	0.417	0.0	0.0	0.0	0.678	0.133	0.0	-0.84
Qwen2-Audio-7B-Instruct	0.712	**0.611**	0.0	0.42	0.374	0.0	0.0	0.68	0.094	0.151	-0.88
Qwen2-Audio-7B-Instruct-cyrl	0.683	0.0	0.0	0.228	0.0	0.0	0.0	0.665	0.097	0.0	-0.82
wav2vec2-large-ipa	0.797	0.389	0.0	**0.468**	**0.586**	0.0	0.0	0.725	0.038	0.261	-0.89
wav2vec2-large-ipa-zs	0.326	0.0	0.0	0.0	0.0	0.0	0.0	0.309	0.22	0.0	-0.77
Qwen2.5-Omni-7B	0.693	0.278	0.0	0.28	0.246	0.0	0.0	0.642	0.071	0.024	-0.9
Qwen2.5-Omni-7B-cyrl	0.686	0.0	0.0	0.199	0.0	0.0	0.0	0.607	0.107	0.0	-0.82
w2v2l-custom-avg	0.79	0.222	0.0	0.36	0.567	0.0	0.0	**0.748**	0.107	0.315	-0.92
w2v2l-custom-avg-zs	0.415	0.0	0.0	0.0	0.0	0.0	0.0	0.331	0.0	0.0	-0.78
w2v2l-custom	0.757	0.222	0.0	0.356	0.479	0.0	0.0	0.699	0.133	0.263	-0.94
w2v2l-custom-avg-lm	0.789	0.333	0.0	0.363	0.567	0.0	0.0	0.747	0.074	0.32	-0.92
w2v2l-custom-cpy1	**0.809**	0.317	0.0	0.389	0.506	0.0	**0.667**	0.732	**0.138**	**0.35**	-0.66

Table 4: Phoneme category-wise averaged F1 scores and their Pearson's correlation coefficients r with complexities (length) - Kina Rutul

level variability, with R^2 values typically in the 0.45–0.70 range.

This pattern implies that many phonemes traditionally described as "complex"—including labialized, ejective, and pharyngealized segments—are difficult for ASR systems primarily because they are rare in the training data. Figure 1 illustrates this effect for a representative model–language pair, where each point corresponds to a phoneme, the horizontal axis denotes log training frequency, and point size reflects log (test frequency+1). The fitted sigmoid captures the dominant trend, while deviations reveal model- and language-specific effects.

Notable deviations from the sigmoid trend occur for whisper-large-v3 and wav2vec2-large-ipa on Archi, where several extremely low-frequency phonemes achieve higher F1 scores than predicted by the fitted curve. These cases are characterized by weaker overall fits (lower R^2), suggesting few-shot transfer from multilingual pretraining. Importantly, this behavior is not observed for Rutul, where the more conversational and spontaneous speech style (while the Archi corpus is read out, see §3) leads these models to adhere more closely to the average sigmoid behavior.

To further probe these cases, we analyze low-support phonemes ($< 10^{1.6}$) by matching them to high-support counterparts ($> 10^{1.9}$) sharing the same base segment. Many well-performing low-support phonemes (F1 > 0.8) are systematic diacritic variants of frequent bases (e.g., aː/a^{ˤ} → a; eː/e^{ˤ} → e; oː/o^{ˤ} → o; ʃː/ʃː^{w} → ʃ), with moderate correlation ($\rho \sim 0.4 - 0.49$). However, some consonants (e.g., g^{w}, k^{w}, t͡ʃ') are not captured by this matching, suggesting that similarity-based effects may partly explain the deviations, while transfer from pretrained representations remains a plausible but inconclusive factor.

Zero-shot systems—wav2vec2-large-ipa-zs, w2v2l-custom-avg-zs, and gpt-4o-transcribe—also exhibit sigmoid-shaped trends, but with transition regions shifted toward higher training frequencies (larger $\log x_0$). As a result, even moderately frequent phonemes fall within the low-accuracy regime, highlighting that language-specific fine-tuning primarily acts to shift the sigmoid leftward, enabling effective learning at substantially lower levels of phoneme support.

Consistent with this interpretation, the estimated sigmoid midpoints ($\log x_0$) cluster around 1.6 ± 0.3 for Archi and 2.1 ± 0.4 for Rutul, excluding the anomalous cases. This suggests that achieving on the order of 10^2 training instances per phoneme may be sufficient to move most segments into the steep learning regime, providing a practical target for future data collection efforts. The right-shift in Rutul's estimated midpoint may partly reflect differences in speech conditions, as it consists of spontaneous speech recorded in noisier settings with greater speaker variability compared to the read speech in Archi (see §3).

Overall, the frequency–F1 sigmoid provides a unifying explanation for phoneme-level error patterns across models.

6.4 Qualitative Error Analysis

Figure 3 shows normalized phoneme confusion matrices for the w2v2l-custom-avg model on Archi and Kina Rutul. Both matrices are strongly diagonal, indicating that most phonemes are correctly recognized, with errors concentrated in a small number of systematic confusions.

A recurring pattern in both languages is the reduction of marked phonemes to their unmarked

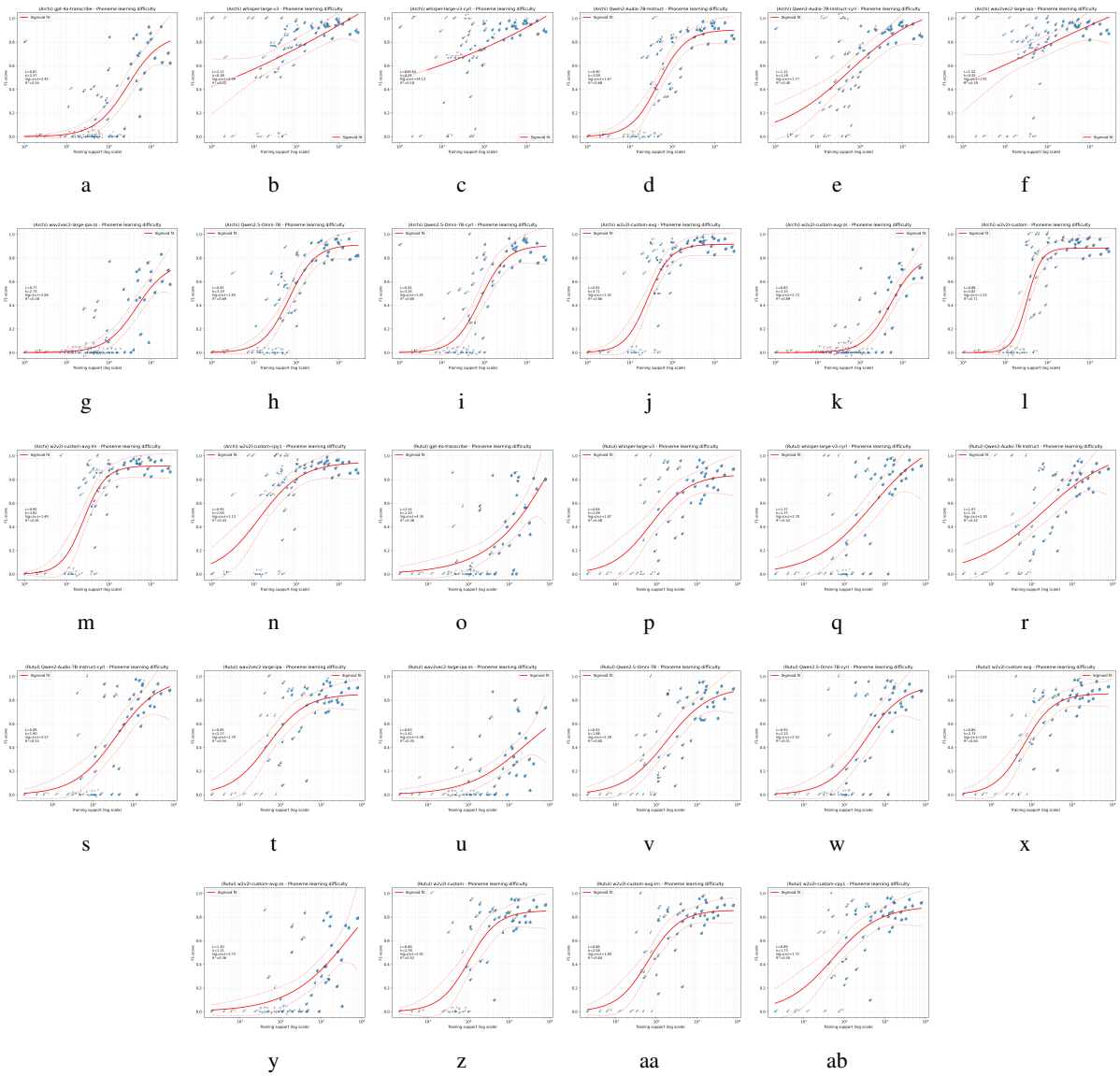

Figure 2: Phoneme-level F1-scores against (log) training frequencies, illustrating a characteristic sigmoid-shaped learning trend in most cases. Exceptions are zero-shot applications (a, g, k, o, u, y) and cases demonstrating few-shot capabilities (b, c).

counterparts. For example, long and pharyngealized vowels such as /iː/ and /iˤ/ are frequently confused with the short vowel /i/. Similar effects are observed for consonants with secondary articulations, where labialized or pharyngealized segments are often mapped to their plain counterparts.

Other frequent error types include vowel quality confusions and incorrect word boundary detection. Examples (1) from Archi and (2) from Kina Rutul compare expert transcriptions with ASR outputs from the best-performing models (whisper-large-v3 for Archi and w2v2l-custom-avg for Kina Rutul).

(1) Archi example

Expert: os t͡ʃ'emna os boʃormi̲n halma̲χdu̲ e̲wdili jatːiki̲ ʟ'arak war ʟirtːu ikʲ'ʷ wi̲χdu hibatːu

ASR: os t͡ʃ'emna os boʃormu̲n hal ma̲χd i̲wdili jatːi_ ʟ'arak war ʟirtːu iki̲ u̲iχdu hibatːu

Translation: "There was once a man who had a friend, one who would go all the way down for you, with a trustful heart, a good one."

This example illustrates a case of incorrect word segmentation (*halmaχdu*), attributable to ASR. Likewise, the output *uiχdu* instead of *wiχdu*, and the realization of /kʲ'ʷ/ as /k/, plausibly reflect frequency-driven confusions during training. In contrast, substitutions such as /i/ → /u/ in *boʃormin* and /e/ → /i/ in *ewdili* may be influenced by variation between the consultants' pronunciation and

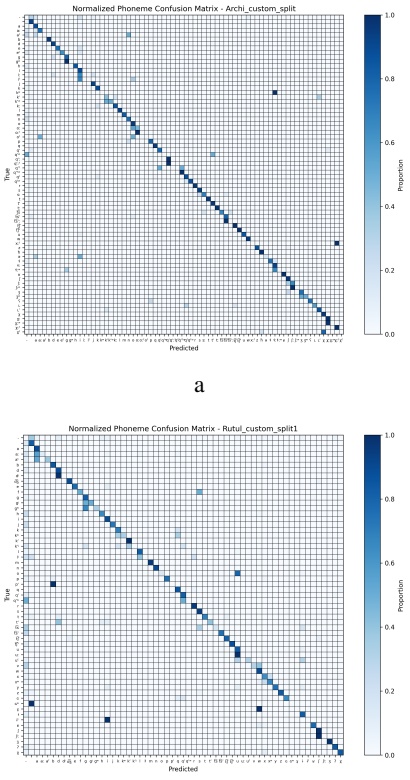

Figure 3: Phoneme confusion matrices of model w2v2l-custom-avg on (a) Archi and (b) Kina Rutul

the orthography used in the texts they were reading, for instance in the oblique stem morpheme -mu- ∼ -mi- or in the masculine verb form *ewdi* ∼ *iwdi*.

(2) Kina Rutul example
Expert: mu$^{\text{ʕ}}$ ɢ$^{\text{w}}$aː ɤiselit daʔatʼij kar haʔatʼij
ASR: muqʼ$^{\text{w}}$oʕ ɯisi ritdaʔadiqar had i
Translation: "They were ousting (people) from the village, they were doing things like this."

Here, the rare phoneme ɢ is confused with the much more frequent qʼ, illustrating a typical frequency-driven substitution. Incorrect word boundary detection is also evident and occurs more frequently than in Archi, consistent with the more spontaneous and conversational nature of the Rutul recordings (see §3). Some additional mismatches may be epiphenomenal: pharyngealization is prosodic rather than strictly segmental, so that its displacement from the root to the final vowel is not phonetically unmotivated and cannot be blamed on ASR. Similarly, the realization of intervocalic glottal stops (e.g., in *daʔat*) may vary due to phonetic weakening in casual speech or transcription conventions.

7 Discussion

Our analysis highlights the central role of frequency in phoneme recognition and connects, at a high level, to related work on frequency-driven learning in lexical processing such as Murray and Forster (2004). Recently, Heitmeier et al. (2024) have shown that word frequency induces an S-shaped relationship with processing difficulty, arising from learning dynamics rather than articulatory complexity. Although their work concerns lexical representations and human reaction times, we observe a qualitatively similar sigmoid relationship between phoneme-level training frequency and ASR accuracy.

In addition, Archi and Rutul differ in their phonological inventories, primarily in the presence of palatalization in Kina Rutul and gemination in Archi (see Appendix A). Despite these differences, the frequency–performance relationship appears consistently in both languages and in model families. This suggests that the observed sigmoid learning dynamics are not tied to a specific phonological system but may reflect more general properties of supervised phoneme learning in low-resource phonologically-complex ASR.

Furthermore, estimated sigmoid midpoints indicate that, excluding anomalous cases, phonemes begin to enter the steep learning regime at roughly 10^2 training instances, suggesting a practical target for prioritizing phoneme coverage during data collection.

8 Conclusion

We present the first phoneme-level ASR analysis for Archi and Kina Rutul, based on curated speech–transcript resources consolidated into a benchmark suitable for ASR training and evaluation. We show that a simple phoneme-vocabulary adaptation with heuristic initialization yields substantial gains for wav2vec2-large-IPA under extremely low-resource conditions.

Our analyses indicate that many errors often attributed to phonological complexity are equally predictable from training frequency. While limited in scale, these findings suggest that frequency-informed perspectives may generalize across distinct phoneme inventories and offer practical guidance for data collection and evaluation in low-resource ASR. At the same time, broader validation across languages and typological settings remains an important direction for future work.

Limitations

This study focuses on two East Caucasian languages and relies on relatively small, manually curated datasets. Although each corpus contains on the order of one hour of transcribed speech, these resources constitute a meaningful first step for Archi and Kina Rutul—endangered languages with exceptionally rich phonological systems—for which no prior ASR benchmarks or training-ready resources existed. To our knowledge, East Caucasian languages are least likely to be present in widely used pretraining corpora (e.g., Common Voice, FLEURS) (Ardila et al., 2020; Conneau et al., 2023), as we did not identify any such languages in these datasets. The phoneme-level results for gpt-4o-transcribe, where several phonemes are not recognized at all (Figure 2 a & o), are consistent with limited direct transfer, though the exact composition of pretraining data remains unknown.

Our analyses operate in an extreme low-resource regime, where phoneme-level effects become visible precisely because data are sparse. In this setting, we observe systematic relationships between phoneme-level training frequency and recognition performance. While such fine-grained effects may attenuate or change at larger data scales, they offer insight into ASR behavior at resolutions that are rarely accessible in higher-resource settings. That said, we do not claim direct transferability of these observations to other language families, data regimes, or representational units.

In addition, our operationalization of phoneme complexity—approximated via diacritic length—serves only as a coarse proxy. Future work could incorporate more principled articulatory or typological measures, for example by considering cross-linguistic phoneme frequencies or articulatory feature inventories.

Ethics Statement

The speech recordings and transcriptions used in this study were obtained with permission from the original authors of the respective documentation projects and were used solely for research purposes. All speakers were recorded with informed consent as part of those projects.

Large language models, including ChatGPT, were used as assistive tools during the preparation of code and manuscript text. Their use was limited to generating small, self-contained code or writing suggestions, and all such outputs were carefully reviewed, corrected, and validated by the authors. ChatGPT did not contribute original scientific ideas, analyses, or results.

Acknowledgments

This research utilized compute resources at the Tübingen Machine Learning Cloud, DFG FKZ INST 37/1057-1 FUGG. Mahesh Akavarapu received funding from Volkswagen Foundation under the Phylomilia project within the Pioneering Projects funding line.

References

Anastasia Alekseeva, Nikita Beklemishev, Michael Daniel, Nina Dobrushina, Konstantin Filatov, Anastasia Ivanova, Timur Maisak, and Ivan Osorgin. 2024. Dictionary of kina rutul.

Rosana Ardila, Megan Branson, Kelly Davis, Michael Kohler, Josh Meyer, Michael Henretty, Reuben Morais, Lindsay Saunders, Francis Tyers, and Gregor Weber. 2020. Common voice: A massively-multilingual speech corpus. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4218–4222, Marseille, France. European Language Resources Association.

Vipul Arora, Aditi Lahiri, and Henning Reetz. 2018. Phonological feature-based speech recognition system for pronunciation training in non-native language learning. *The Journal of the Acoustical Society of America*, 143(1):98–108.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, and 1 others. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Gilles Boulianne. 2022. Phoneme transcription of endangered languages: an evaluation of recent ASR architectures in the single speaker scenario. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2301–2308, Dublin, Ireland. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng

He, Junyang Lin, and 1 others. 2024. Qwen2-audio technical report. *arXiv preprint arXiv:2407.10759*.

Marina Chumakina, Dunstan Brown, Greville Corbett, and Harley Quilliam. 2007. A dictionary of archi: Archi-russian-english. *University of Surrey*.

Alexis Conneau, Min Ma, Simran Khanuja, Yu Zhang, Vera Axelrod, Siddharth Dalmia, Jason Riesa, Clara Rivera, and Ankur Bapna. 2023. Fleurs: Few-shot learning evaluation of universal representations of speech. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 798–805. IEEE.

Seraphina Fong, Marco Matassoni, Alessio Brutti, and 1 others. 2025. Speech llms in low-resource scenarios: Data volume requirements and the impact of pretraining on high-resource languages. In *Proceedings of Interspeech*, pages 2003–2007. ISCA-International Speech Communication Association.

Xuelong Geng, Kun Wei, Qijie Shao, Shuiyun Liu, Zhennan Lin, Zhixian Zhao, Guojian Li, Wenjie Tian, Peikun Chen, Yangze Li, and 1 others. 2025. Osum: Advancing open speech understanding models with limited resources in academia. *arXiv preprint arXiv:2501.13306*.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376.

Séverine Guillaume, Guillaume Wisniewski, Benjamin Galliot, Minh-Châu Nguyễn, Maxime Fily, Guillaume Jacques, and Alexis Michaud. 2022. Plugging a neural phoneme recognizer into a simple language model: a workflow for low-resource settings. In *Interspeech 2022-23rd Annual Conference of the International Speech Communication Association*, pages 4905–4909. International Speech Communication Association.

Kenneth Heafield. 2011. KenLM: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197, Edinburgh, Scotland. Association for Computational Linguistics.

Maria Heitmeier, Yu-Ying Chuang, Seth D Axen, and R Harald Baayen. 2024. Frequency effects in linear discriminative learning. *Frontiers in Human Neuroscience*, 17:1242720.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Aleksandr E. Kibrik, Sandro V. Kodzasov, Irina P. Olovyannikova, Dzhalil S. Samedov, Michael Daniel, Anna Khoroshkina, and Alexandre Arkhipov. 2007. Archi text corpus (1.0).

Aleksandr E Kibrik, SV Kodzasov, IP Olovyannikova, and DS Samedov. 1977. Opyt strukturnogo opisaniya archinskogo yazyka. *TI Leksika. Ponetika [Essay on a Structural Description of the Artchi Language, Vol. 1: Lexis. Phonetics]. Moscow*.

Diederik P Kingma. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Xinjian Li, Florian Metze, David R Mortensen, Alan W Black, and Shinji Watanabe. 2022. Asr2k: Speech recognition for around 2000 languages without audio. In *Proc. Interspeech 2022*, pages 4885–4889.

Zhaolin Li and Jan Niehues. 2025a. Enhance contextual learning in ASR for endangered low-resource languages. In *Proceedings of the 1st Workshop on Language Models for Underserved Communities (LM4UC 2025)*, pages 1–7, Albuquerque, New Mexico. Association for Computational Linguistics.

Zhaolin Li and Jan Niehues. 2025b. In-context language learning for endangered languages in speech recognition. In *Proceedings of Interspeech*.

Siyu Liang and Gina-Anne Levow. 2025. Breaking the transcription bottleneck: Fine-tuning ASR models for extremely low-resource fieldwork languages. In *Proceedings of the Fourth Workshop on NLP Applications to Field Linguistics*, pages 26–37, Vienna, Austria. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Donald W Marquardt. 1963. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the society for Industrial and Applied Mathematics*, 11(2):431–441.

Wayne S Murray and Kenneth I Forster. 2004. Serial mechanisms in lexical access: the rank hypothesis. *Psychological Review*, 111(3):721.

Vineel Pratap, Andros Tjandra, Bowen Shi, Paden Tomasello, Arun Babu, Sayani Kundu, Ali Elkahky, Zhaoheng Ni, Apoorv Vyas, Maryam Fazel-Zarandi, and 1 others. 2024. Scaling speech technology to 1,000+ languages. *Journal of Machine Learning Research*, 25(97):1–52.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR.

Kathleen Siminyu, Xinjian Li, Antonios Anastasopoulos, David R Mortensen, Michael R Marlo, and Graham Neubig. 2021. Phoneme recognition through

fine tuning of phonetic representations: A case study on luhya language varieties. In *Proc. Interspeech 2021*, pages 271–275.

Chihiro Taguchi, Yusuke Sakai, Parisa Haghani, and David Chiang. 2023. Universal automatic phonetic transcription into the international phonetic alphabet. In *Proceedings of Interspeech*.

Qwen Team and 1 others. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2(3).

Aad W Van der Vaart. 2000. *Asymptotic statistics*, volume 3. Cambridge university press.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, and 1 others. 2025. Qwen2. 5-omni technical report. *arXiv preprint arXiv:2503.20215*.

Qiantong Xu, Alexei Baevski, and Michael Auli. 2022. Simple and effective zero-shot cross-lingual phoneme recognition. In *Proc. Interspeech 2022*, pages 2113–2117.

Saierdaer Yusuyin, Te Ma, Hao Huang, Wenbo Zhao, and Zhijian Ou. 2025. Whistle: Data-efficient multilingual and crosslingual speech recognition via weakly phonetic supervision. *IEEE Transactions on Audio, Speech and Language Processing*.

Appendix

A IPA-Cyrillic map

This section provides the mapping used to convert IPA texts into Cyrillic for employing in models like gpt-4o-transcribe or *-cyrl models. All possible phonemes occurring for each language in the datasets are also covered here. Note that a few phonemes that occur in only Russian borrowings are nevertheless included in our vocabulary and listed here. In the case of absence of one-one mapping for a phoneme due to borrowings, the native phonemes are given preference while converting from Cyrillic to IPA.

Archi: a-а aː-аа aːˤ-ааI aˤ-аI b-б d-д e-е eː-ее eːˤ-ееI eˤ-еI g-г gʷ-гв h-гь i-и iː-ии iˤ-иI j-й k-к kʷ-кв kʼ-кI kʼʷ-кIв kː-кк kːʷ-ккв l-л m-м n-н o-о oː-оо oːˤ-ооI oˤ-оI p-п pʼ-пI pː-пп q-хъ qʷ-хъв qʼ-къ qʼʷ-къв qʼː-ккъ qʼːˤ-ккъI qʼˤ-къI qʼˤʷ-къIв qˤ-хъI qˤʷ-хъIв r-р s-с sː-сс t-т tʼ-тI tː-тт t͡s-ц t͡sʼ-цI t͡sʼː-ццI t͡ʃ-ч t͡ʃʼ-чI u-у uː-уу uˤ-уI w-в z-з ħ-хI ə-ы ɬ-лъ ɬː-ллъ ɬːʷ-ллъв ʁ-гъ ʁˤ-гъI ʃ-ш ʃʷ-шв ʃː-щ ʃːʷ-щв ʒ-ж ʒʷ-жв ʔ-ъ ʕ-гI ʟ-лI ʟʼ-кь ʟʼʷ-кьв χ-х χʷ-хв χː-хх χːʷ-ххв χːˤ-ххьI χˤ-хьI

Rutul: a-а aː-аа aˤ-аI b-б d-д dʲ-дʼ d͡ʒ-дж d͡ʒʷ-джв e-е eˤ-еI f-ф g-г gʲ-гʼ gʷ-гв h-гь hʷ-гьв i-и iː-ии iˤ-иI j-й k-к kʲ-кʼ kʷ-кв kʼ-кI kʼʲ-кIʼ kʼʷ-кIв l-л lʲ-лʼ m-м mʲ-мʼ n-н nʲ-нʼ o-о oˤ-оI p-п pʲ-пʼ pʼ-пI q-хъ qʷ-хъв qʼ-хъI qʼʷ-хъIв r-р s-с sʲ-сʼ t-т tʲ-тʼ tʼ-тI t͡s-ц t͡sʼ-цI t͡ʃ-ч t͡ʃʷ-чв t͡ʃʼ-чI u-у uː-уу uˤ-уI w-в wʲ-вʼ x-хь xʲ-хьʼ xʷ-хьв y-уь z-з ø-ё ɢ-къ ɢʷ-къв ɣ-гI ɨ-ы ɨː-ыы ɨˤ-ыI ʁ-гъ ʁʷ-гъв ʃ-ш ʃː-щ ʒ-ж ʔ-ъ χ-х χʷ-хв

B Wilcoxon Signed test: p-values

This section provides p-values to compare the scores of the main results (Table 2) for each language.

B.1 Archi

Tests on CER values

	w2v2l-custom -avg	w2v2l-custom -avg-lm	w2v2l-custom -cpy1	w2v2l-custom	wav2vec2 -large-ipa	whisper -large-v3	Qwen2-Audio -7B-Instruct	Qwen2.5 -Omni-7B
w2v2l-custom-avg (ours)	-	0.901	0.810	<1e-3	<1e-3	0.170	<1e-3	<1e-3
w2v2l-custom-avg-lm (ours)	0.901	-	0.847	0.048	0.059	0.049	<1e-3	<1e-3
w2v2l-custom-cpy1	0.810	0.847	-	0.078	0.117	0.151	<1e-3	<1e-3
w2v2l-custom	<1e-3	0.048	0.078	-	0.510	0.001	0.027	<1e-3
wav2vec2-large-ipa	<1e-3	0.059	0.117	0.510	-	0.002	0.029	<1e-3
whisper-large-v3	0.170	0.049	0.151	0.001	0.002	-	<1e-3	<1e-3
Qwen2-Audio-7B-Instruct	<1e-3	<1e-3	<1e-3	0.027	0.029	<1e-3	-	0.002
Qwen2.5-Omni-7B	<1e-3	<1e-3	<1e-3	<1e-3	<1e-3	<1e-3	0.002	-

Tests on WER values

	w2v2l-custom -avg	w2v2l-custom -avg-lm	w2v2l-custom -cpy1	w2v2l-custom	wav2vec2 -large-ipa	whisper -large-v3	Qwen2-Audio -7B-Instruct	Qwen2.5 -Omni-7B
w2v2l-custom-avg (ours)	-	0.465	0.641	<1e-3	<1e-3	0.039	0.005	<1e-3
w2v2l-custom-avg-lm (ours)	0.465	-	0.928	<1e-3	<1e-3	0.036	<1e-3	<1e-3
w2v2l-custom-cpy1	0.641	0.928	-	<1e-3	0.001	0.079	0.002	<1e-3
w2v2l-custom	<1e-3	<1e-3	<1e-3	-	0.891	<1e-3	0.822	<1e-3
wav2vec2-large-ipa	<1e-3	<1e-3	0.001	0.891	-	<1e-3	0.919	<1e-3
whisper-large-v3	0.039	0.036	0.079	<1e-3	<1e-3	-	<1e-3	<1e-3
Qwen2-Audio-7B-Instruct	0.005	<1e-3	0.002	0.822	0.919	<1e-3	-	<1e-3
Qwen2.5-Omni-7B	<1e-3	<1e-3	<1e-3	<1e-3	<1e-3	<1e-3	<1e-3	-

B.2 Kina Rutul

Tests on CER values

	w2v2l-custom -avg	w2v2l-custom -avg-lm	w2v2l-custom -cpy1	w2v2l-custom	wav2vec2 -large-ipa	whisper -large-v3	Qwen2-Audio -7B-Instruct	Qwen2.5 -Omni-7B
w2v2l-custom-avg (ours)	-	0.638	0.801	<1e-3	<1e-3	0.013	0.003	<1e-3
w2v2l-custom-avg-lm (ours)	0.638	-	0.737	0.038	0.067	<1e-3	<1e-3	<1e-3
w2v2l-custom-cpy1	0.801	0.737	-	0.112	0.076	0.040	0.010	<1e-3
w2v2l-custom	<1e-3	0.038	0.112	-	0.957	0.458	0.373	0.020
wav2vec2-large-ipa	<1e-3	0.067	0.076	0.957	-	0.505	0.325	0.012
whisper-large-v3	0.013	<1e-3	0.040	0.458	0.505	-	0.679	0.014
Qwen2-Audio-7B-Instruct	0.003	<1e-3	0.010	0.373	0.325	0.679	-	0.002
Qwen2.5-Omni-7B	<1e-3	<1e-3	<1e-3	0.020	0.012	0.014	0.002	-

Tests on WER values

	w2v2l-custom -avg	w2v2l-custom -avg-lm	w2v2l-custom -cpy1	w2v2l-custom	wav2vec2 -large-ipa	whisper -large-v3	Qwen2-Audio -7B-Instruct	Qwen2.5 -Omni-7B
w2v2l-custom-avg (ours)	-	0.157	0.706	0.067	0.006	0.342	0.107	0.002
w2v2l-custom-avg-lm (ours)	0.157	-	0.173	0.007	<1e-3	<1e-3	0.003	<1e-3
w2v2l-custom-cpy1	0.706	0.173	-	0.307	0.041	0.185	0.135	0.001
w2v2l-custom	0.067	0.007	0.307	-	0.281	0.747	0.955	0.072
wav2vec2-large-ipa	0.006	<1e-3	0.041	0.281	-	0.487	0.761	0.398
whisper-large-v3	0.342	<1e-3	0.185	0.747	0.487	-	0.391	0.013
Qwen2-Audio-7B-Instruct	0.107	0.003	0.135	0.955	0.761	0.391	-	0.047
Qwen2.5-Omni-7B	0.002	<1e-3	0.001	0.072	0.398	0.013	0.047	-

C Quality vs Quantity in Rutul

In the case of Kina Rutul, the set split2 contains good quality speech (as judged by the expert, i.e., the second author), however consist only 394 sentences (utterances) while split3 contains speech of acceptable quality with 994 sentences. The evaluations on each of these splits is tabulated here here.

Model	Split	WER	CER	PER
w2v2l-custom-avg	split2	0.891	0.264	0.252
w2v2l-custom-avg	split3	0.783	**0.216**	**0.213**
wav2vec2-large-ipa	split2	0.880	0.271	0.259
wav2vec2-large-ipa	split3	**0.782**	0.226	0.223

D Phonemes with least F1 scores

The top-10 most challenging phonemes for each model and language are listed here. Each triplet represents (<phoneme>, F1 score, test frequency).

D.1 Archi

Model	1	2	3	4	5	6	7	8	9	10
w2v2l-custom-avg	(gʷ, 0.0, 6)	(iː, 0.0, 6)	(χˤ, 0.0, 5)	(aˤ, 0.0, 4)	(oːˤ, 0.0, 4)	(iˤ, 0.0, 3)	(ɬːʷ, 0.0, 3)	(oˤ, 0.0, 2)	(qʼʷ, 0.0, 2)	(qʼːˤ, 0.0, 2)
w2v2l-custom-avg-lm	(gʷ, 0.0, 6)	(iː, 0.0, 6)	(χˤ, 0.0, 5)	(aˤ, 0.0, 4)	(oːˤ, 0.0, 4)	(iˤ, 0.0, 3)	(ɬːʷ, 0.0, 3)	(oˤ, 0.0, 2)	(qʼʷ, 0.0, 2)	(qʼːˤ, 0.0, 2)
w2v2l-custom-cpy1	(gʷ, 0.0, 6)	(iː, 0.0, 6)	(χˤ, 0.0, 5)	(aˤ, 0.0, 4)	(iˤ, 0.0, 3)	(qʼʷ, 0.0, 2)	(qʼˤ, 0.0, 2)	(kʷ, 0.0, 1)	(t͡sʼː, 0.0, 1)	(xːˤ, 0.0, 1)
w2v2l-custom	(eˤ, 0.0, 8)	(gʷ, 0.0, 6)	(iː, 0.0, 6)	(t͡s, 0.0, 6)	(ɔ, 0.0, 6)	(qʼː, 0.0, 5)	(χˤ, 0.0, 5)	(aˤ, 0.0, 4)	(oːˤ, 0.0, 4)	(iˤ, 0.0, 3)
w2v2l-custom-avg-zs	(tː, 0.0, 89)	(χ, 0.0, 78)	(tʼ, 0.0, 57)	(ɬː, 0.0, 37)	(qˤ, 0.0, 23)	(q, 0.0, 21)	(kʼ, 0.0, 17)	(Lʼ, 0.0, 15)	(t͡ʃʼ, 0.0, 14)	(ɬ, 0.0, 12)
gpt-4o-transcribe	(h, 0.0, 42)	(ɬː, 0.0, 37)	(qˤ, 0.0, 23)	(q, 0.0, 21)	(kʼ, 0.0, 17)	(Lʼ, 0.0, 15)	(t͡ʃʼ, 0.0, 14)	(ɬ, 0.0, 12)	(t͡sʼ, 0.0, 10)	(eˤ, 0.0, 8)
Qwen2-Audio-7B-Instruct	(eˤ, 0.0, 8)	(gʷ, 0.0, 6)	(iː, 0.0, 6)	(ɔ, 0.0, 6)	(χˤ, 0.0, 5)	(aˤ, 0.0, 4)	(oːˤ, 0.0, 4)	(iˤ, 0.0, 3)	(ɬːʷ, 0.0, 3)	(kʼʷ, 0.0, 2)
Qwen2-Audio-7B-Instruct-cyrl	(ɔ, 0.0, 6)	(χˤ, 0.0, 5)	(aˤ, 0.0, 4)	(iˤ, 0.0, 3)	(ʃː, 0.0, 3)	(qʼʷ, 0.0, 2)	(qʼːˤ, 0.0, 2)	(ʒʷ, 0.0, 2)	(ʃːʷ, 0.0, 1)	(χːʷ, 0.0, 1)
Qwen2.5-Omni-7B	(eˤ, 0.0, 8)	(L, 0.0, 7)	(χːˤ, 0.0, 7)	(gʷ, 0.0, 6)	(ɔ, 0.0, 6)	(χˤ, 0.0, 5)	(aˤ, 0.0, 4)	(oːˤ, 0.0, 4)	(iˤ, 0.0, 3)	(qʼˤʷ, 0.0, 3)
Qwen2.5-Omni-7B-cyrl	(kʼ, 0.0, 17)	(Lʼ, 0.0, 15)	(eˤ, 0.0, 8)	(χːˤ, 0.0, 7)	(ɔ, 0.0, 6)	(qʼː, 0.0, 5)	(ɦ, 0.0, 5)	(χˤ, 0.0, 5)	(aˤ, 0.0, 4)	(oːˤ, 0.0, 4)
wav2vec2-large-ipa	(iː, 0.0, 6)	(aˤ, 0.0, 4)	(iˤ, 0.0, 3)	(qʼʷ, 0.0, 2)	(qʼːˤ, 0.0, 2)	(ʒʷ, 0.0, 2)	(t͡sʼː, 0.0, 1)	(xːˤ, 0.0, 1)	(ɔ, 0.15, 6)	(qʼˤ, 0.33, 2)
wav2vec2-large-ipa-zs	(χ, 0.0, 78)	(tʼ, 0.0, 57)	(ɬː, 0.0, 37)	(qˤ, 0.0, 23)	(q, 0.0, 21)	(kʼ, 0.0, 17)	(Lʼ, 0.0, 15)	(t͡ʃʼ, 0.0, 14)	(ɬ, 0.0, 12)	(sː, 0.0, 10)
whisper-large-v3	(ɔ, 0.0, 6)	(aˤ, 0.0, 4)	(iˤ, 0.0, 3)	(kʼʷ, 0.0, 2)	(qʼːˤ, 0.0, 2)	(qʼˤ, 0.0, 2)	(kʷ, 0.0, 1)	(t͡sʼː, 0.0, 1)	(iː, 0.5, 6)	(ʁ, 0.5, 4)
whisper-large-v3-cyrl	(ɔ, 0.0, 6)	(h, 0.0, 5)	(χˤ, 0.0, 5)	(iˤ, 0.0, 3)	(qʼːˤ, 0.0, 2)	(qʼˤ, 0.0, 2)	(ʕ, 0.06, 4)	(iː, 0.29, 6)	(aˤ, 0.33, 4)	(kʼ, 0.54, 17)

D.2 Kina Rutul

Model	1	2	3	4	5	6	7	8	9	10
w2v2l-custom-avg	(kʼʲ, 0.0, 6)	(lʲ, 0.0, 5)	(qʼʷ, 0.0, 2)	(uː, 0.0, 2)	(ʃjː, 0.0, 2)	(dʲ, 0.0, 1)	(pʼ, 0.0, 1)	(Gʷ, 0.0, 1)	(ɣ, 0.0, 1)	(iˤ, 0.0, 1)
w2v2l-custom-avg-lm	(kʼʲ, 0.0, 6)	(qʼʷ, 0.0, 2)	(uː, 0.0, 2)	(βʲː, 0.0, 2)	(dʲ, 0.0, 1)	(pʼ, 0.0, 1)	(Gʷ, 0.0, 1)	(ɣ, 0.0, 1)	(iˤ, 0.0, 1)	(aː, 0.15, 25)
w2v2l-custom-cpy1	(kʼʲ, 0.0, 6)	(uː, 0.0, 2)	(βʲː, 0.0, 2)	(dʲ, 0.0, 1)	(kʼ, 0.0, 1)	(pʼ, 0.0, 1)	(Gʷ, 0.0, 1)	(iˤ, 0.0, 1)	(o, 0.1, 35)	(v, 0.22, 10)
w2v2l-custom	(v, 0.0, 10)	(kʼʲ, 0.0, 6)	(t͡sˤ, 0.0, 6)	(lʲ, 0.0, 5)	(gʷ, 0.0, 3)	(kʷ, 0.0, 3)	(qʼʷ, 0.0, 2)	(uː, 0.0, 2)	(βʲː, 0.0, 2)	(dʲ, 0.0, 1)
w2v2l-custom-avg-zs	(χ, 0.0, 72)	(xʷ, 0.0, 58)	(ʁ, 0.0, 45)	(G, 0.0, 32)	(qʼ, 0.0, 27)	(aː, 0.0, 25)	(aˤ, 0.0, 21)	(q, 0.0, 21)	(uˤ, 0.0, 13)	(y, 0.0, 12)
gpt-4o-transcribe	(h, 0.0, 139)	(ʕ, 0.0, 89)	(xʷ, 0.0, 58)	(ʁ, 0.0, 45)	(qʼ, 0.0, 27)	(aˤ, 0.0, 21)	(q, 0.0, 21)	(uˤ, 0.0, 13)	(y, 0.0, 12)	(tʼ, 0.0, 10)
Qwen2-Audio-7B-Instruct	(kʼʲ, 0.0, 6)	(gʷ, 0.0, 3)	(qʼʷ, 0.0, 2)	(uː, 0.0, 2)	(βʲː, 0.0, 2)	(kʼ, 0.0, 1)	(pʼ, 0.0, 1)	(ɣ, 0.0, 1)	(iˤ, 0.0, 1)	(uˤ, 0.11, 13)
Qwen2-Audio-7B-Instruct-cyrl	(ʕ, 0.0, 89)	(qʼ, 0.0, 27)	(aˤ, 0.0, 21)	(uˤ, 0.0, 13)	(tʼ, 0.0, 10)	(v, 0.0, 10)	(t͡sʼ, 0.0, 8)	(t͡ʃʼ, 0.0, 8)	(kʼʲ, 0.0, 6)	(lʲ, 0.0, 5)
Qwen2.5-Omni-7B	(uˤ, 0.0, 13)	(kʼʲ, 0.0, 6)	(qʼʷ, 0.0, 2)	(uː, 0.0, 2)	(βʲː, 0.0, 2)	(dʲ, 0.0, 1)	(kʼ, 0.0, 1)	(pʼ, 0.0, 1)	(Gʷ, 0.0, 1)	(ɣ, 0.0, 1)
Qwen2.5-Omni-7B-cyrl	(ʕ, 0.0, 89)	(qʼ, 0.0, 27)	(aˤ, 0.0, 21)	(uˤ, 0.0, 13)	(y, 0.0, 12)	(tʼ, 0.0, 10)	(v, 0.0, 10)	(t͡sʼ, 0.0, 8)	(t͡ʃʼ, 0.0, 8)	(kʼʲ, 0.0, 6)
wav2vec2-large-ipa	(kʼʲ, 0.0, 6)	(qʼʷ, 0.0, 2)	(uː, 0.0, 2)	(βʲː, 0.0, 2)	(dʲ, 0.0, 1)	(pʼ, 0.0, 1)	(Gʷ, 0.0, 1)	(iˤ, 0.0, 1)	(aː, 0.08, 25)	(o, 0.12, 35)
wav2vec2-large-ipa-zs	(χ, 0.0, 72)	(xʷ, 0.0, 58)	(ʁ, 0.0, 45)	(G, 0.0, 32)	(qʼ, 0.0, 27)	(aˤ, 0.0, 21)	(q, 0.0, 21)	(uˤ, 0.0, 13)	(y, 0.0, 12)	(tʼ, 0.0, 10)
whisper-large-v3	(tʼ, 0.0, 10)	(kʼʲ, 0.0, 6)	(kʷ, 0.0, 3)	(qʼʷ, 0.0, 2)	(uː, 0.0, 2)	(βʲː, 0.0, 2)	(dʲ, 0.0, 1)	(kʼ, 0.0, 1)	(pʼ, 0.0, 1)	(Gʷ, 0.0, 1)
whisper-large-v3-cyrl	(j, 0.0, 89)	(qʼ, 0.0, 27)	(aˤ, 0.0, 21)	(uˤ, 0.0, 13)	(tʼ, 0.0, 10)	(v, 0.0, 10)	(t͡sʼ, 0.0, 8)	(t͡ʃʼ, 0.0, 8)	(kʼʲ, 0.0, 6)	(lʲ, 0.0, 5)