Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Speech Representation System For The Karakalpak Automatic Speech Recognition

Niyetbay Uteuliev
Nukus State Technical University
Jabbar Kudaybergenov
Nukus State Technical University
Tangirbergen Kudaybergenov
Nukus State Technical University
Avazjon Marakhimov
Tashkent State Technical University
Alpamis Kutlimuratov
Kimyo International University in Tashkent
Kabul Khudaybergenov
q.xudaybergenov@kiut.uz

Kimyo International University in Tashkent

Article

Keywords:

Posted Date: July 2nd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-9794382/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Speech Representation System For The Karakalpak Automatic Speech Recognition

Niyetbay Uteuliev[1], Jabbar Kudaybergenov[1], Tangirbergen Kudaybergenov[1], Avazjon Marakhimov[2], Alpamis Kutlimuratov[3], and Kabul Khudaybergenov[3,*]

[1]Department of Software Engineering, Nukus State Technical University, Nukus,100147 Uzbekistan
[2]Department of Information Processing and Management Systems, Tashkent State Technical University, Tashkent, 100174, Uzbekistan
[3]Department of Applied Informatics, Kimyo International University in Tashkent, Tashkent, 100121, Uzbekistan
[*]q.xudaybergenov@kiut.uz

ABSTRACT

Recently, automatic speech recognition (ASR) has moved from deeplearning systems toward end-to-end (E2E) neural architectures. E2E models obtain strong results for high-resource languages, but training reliable systems remains difficult for low-resource languages because labeled speech is expensive to collect and verify. Self-supervised learning (SSL) has reduced this dependency by learning speech representations from large unlabeled corpora and then adapting them to a target language with a smaller amount of transcribed data. In this paper, we present a Karakalpak ASR system based on multilingual speech representations. The work follows the wav2vec 2.0 XLS-R approach, and the Karakalpak-specific acoustic model is trained and fine-tuned by the authors using the Karakalpak Speech Corpus (KSC). KSC is a publicly available benchmark speech-to-text dataset collected by the authors and released through Mendeley Data; it contains 50 hours of predominantly read speech from native Karakalpak speakers. The ASR model is fine-tuned with various KSC data setups, and the effect of the amount of labeled speech on recognition quality is discussed. In addition, word-level n-gram language models are used to improve decoding under low-resource conditions. Recognition error analysis is provided in terms of substitution, insertion, and deletion errors. The 50-hour KSC-trained wav2vec2 XLS-R model gives 21.10% WER and 4.32% CER on a held-out test set, and the experimental design treats this result as a reference point for further benchmarking.

1 Introduction

Automatic speech recognition (ASR) refers to the computational process of converting spoken audio signals into their corresponding textual representations. During the last decade, ASR systems built on E2E architectures have improved substantially through stronger neural architectures, better acoustic representations, and larger training corpora. Recent systems increasingly combine CTC, attention, transducer, convolutional, and Transformer-based encoders to model both local acoustic patterns and long-range linguistic dependencies[3,4,16,25,26]. Nevertheless, ASR systems for low-resource languages remains complex tasks to solve. Existing low-resource ASR strategies, such as transfer learning, multilingual ASR, and meta-learning, rely on the assumption that world languages share useful low-level acoustic and phonetic regularities even when they differ in grammar, writing system, and phonology. Multilingual ASR trains a model with data from many languages, transfer learning adapts parameters learned from data-rich languages, and meta-learning searches for language-independent representations that can be adapted with limited data.

More recent ASR systems use self-supervised speech representations to reduce the demand for labeled audio. Instead of requiring every utterance to be manually transcribed, SSL models learn robust acoustic patterns from large datasets of unlabeled speech and are then fine-tuned on a target dataset. wav2vec 2.0, XLSR/XLS-R, HuBERT, WavLM, data2vec, and w2v-BERT show that masked prediction, contrastive learning, and multilingual pretraining can produce reusable speech encoders for downstream ASR[1,2,10–12,21,22]. This is especially useful for Karakalpak, where open speech technology resources have historically been limited.

Karakalpak is a low-resource Turkic language spoken mainly in the Republic of Karakalpakstan, Uzbekistan. It is closely related to Kazakh and Nogai and has agglutinative morphology, vowel harmony, and a subject-object-verb sentence order typical of many Turkic languages. From an ASR perspective, these properties create long word forms, rich inflection, and many surface variants from a single lexical stem. A language model and a careful text normalization pipeline are therefore important for reducing word-level errors. The problem is not only the number of available speech hours, but also the interaction of dialectal variation, orthographic practice, speaker coverage, recording domain, and the size of normalized text data available for decoding.

The Karakalpak Speech Corpus (KSC) has recently addressed a major resource gap by providing speech recordings in WAV format, manually verified transcriptions in standard Karakalpak Latin orthography, and speaker-independent splits for training, validation, and testing[38]. This makes it suitable for E2E ASR, forced alignment, pronunciation modeling, and acoustic analysis. In comparison with large multilingual corpora such as Common Voice, Multilingual LibriSpeech, Libri-Light, VoxPopuli, and FLEURS, KSC is much smaller, but it is directly targeted at Karakalpak and therefore provides a more reliable foundation for language-specific evaluation[6–8,15,24].

The present study follows the recent direction in which multilingual speech representation models are adapted to languages with limited labeled data. In this work, the Karakalpak-specific ASR model is trained and fine-tuned on KSC, the authors' collected and publicly released dataset. This direction has been strengthened by large-scale weakly supervised and multilingual systems such as Whisper, USM, MMS, and SeamlessM4T[23,31–33]. At the same time, reproducible experimentation requires transparent toolkits, stable recipes, and comparable benchmarks; open frameworks and benchmarks such as ESPnet, SpeechBrain, WeNet, SUPERB, and SpeechStew have made this style of ASR research easier to reproduce[9,13,14,17,18,27].

The main contributions in this paper are as follows:

- We present a Karakalpak ASR system using multilingual self-supervised speech representations.
- We train and fine-tune the Karakalpak-specific wav2vec 2.0 XLS-R ASR model on the authors' Karakalpak Speech Corpus and organize the experiments around different amounts of labeled data.
- We train and apply word-level n-gram language models to improve decoding in a low-resource setting.
- We analyze recognition errors through substitution, insertion, and deletion counts, and discuss the effect of in-domain and out-of-domain evaluation.

In Section 2 we present a literature review of speech recognition approaches and related low-resource work. In Section 3 the methodology is described. Section 4 is dedicated to presenting the computational experiments and some observations. Conclusions of the work is provided in Section 5.

2 Related works

The recent advances in ASR can be attributed to two mutually reinforcing factors: the development of more powerful deep learning architectures and their training on large scale speech datasets. End-to-end systems such as CTC, attention-based encoder-decoder models, Transformer, Conformer, wav2vec 2.0, and Whisper-based recognizers have improved recognition quality in many benchmark tasks. Conformer and ContextNet demonstrated the value of combining local convolutional modeling with global context, while Citrinet, Branchformer, E-Branchformer, Emformer, and Transformer-Transducer variants further explored the trade-off between recognition accuracy, streaming latency, and decoding efficiency[3–5,16,20,25,26]. However, unlike English, Mandarin, and other widely studied languages, many languages have only small transcribed corpora. Manual transcription is expensive, quality control is slow, and data collection often requires community-level linguistic expertise.

Transfer learning and multilingual transfer learning have therefore become central methods for low-resource ASR. In transfer learning, a model trained on a high-resource language is reused and fine-tuned for the target language. In multilingual ASR, several languages are trained together so that shared layers learn acoustic and phonetic regularities that are useful across languages. Meta-learning has also been used to improve adaptation when only a small number of labeled utterances are available. Studies on multilingual and low-resource evaluation have shown that model robustness depends on language coverage, domain match, speaker diversity, and fair evaluation protocols, not only on the number of model parameters[6–8,15,24,29].

Self-supervised pretraining has further changed low-resource ASR. Wav2vec 2.0 learns contextual acoustic representations from raw audio by solving contrastive tasks over masked latent speech frames. XLS-R extends this idea to multilingual speech representation learning by pretraining on speech from many languages. HuBERT, WavLM, data2vec, w2v-BERT, BEATs, and OWSM-style open speech models show several related routes: predicting discrete hidden units, denoising masked speech, distilling contextual representations, combining acoustic and linguistic objectives, or scaling supervised and weakly supervised multilingual data[1,10–12,21,22,34,35]. The pretrained model can then be fine-tuned with a comparatively small amount of transcribed target-language speech. This strategy is useful for languages such as Karakalpak because unlabeled audio is easier to obtain than carefully checked speech-text pairs.

Low-resource ASR studies for Turkic languages show that related-language transfer can be beneficial. Multilingual E2E systems for Kazakh, Bashkir, Kyrgyz, Sakha, and Tatar have demonstrated that shared agglutinative morphology, cognate vocabulary, and common alphabetic patterns can improve performance for critically low-resource languages[39,40]. Kazakh ASR studies using wav2vec 2.0 and related pretrained models are also relevant because Kazakh and Karakalpak share Turkic morphology and regional contact patterns[36,37]. Broader low-resource experiments with wav2vec 2.0 and domain-shifted ASR further indicate that pretrained encoders remain sensitive to recording style, channel conditions, and target-domain

vocabulary[28,30]. Such findings are relevant to Karakalpak because it belongs to the Turkic family and shares structural properties with Kazakh and other neighboring languages.

Recent Karakalpak resources make this line of research practical. The Karakalpak Speech Corpus is a publicly available speech-to-text dataset for Karakalpak ASR and contains approximately 50 hours of predominantly read speech collected by the authors from native Karakalpak speakers with balanced metadata and manually verified Latin-script transcriptions[38]. For the present work, this Mendeley Data resource is treated as the main reliable source for dataset description and experimental design. Its value is that it makes Karakalpak ASR measurable with explicit train, validation, and test organization rather than relying on informal audio collections or unverified model-card claims.

2.1 Karakalpak ASR

Karakalpak is a Turkic language used primarily in Karakalpakstan, with strong contact with Uzbek, Kazakh, and Russian. Its agglutinative morphology forms words by attaching multiple suffixes to a root. This increases the number of distinct word forms and makes word-level modeling difficult when the text corpus is small. Character-level CTC decoding is therefore attractive because it avoids the need for a fixed word vocabulary during acoustic training, while a word-level language model can still be added during inference to prefer fluent Karakalpak word sequences.

Karakalpak uses Latin and Cyrillic writing practices, while the current benchmark corpus provides transcriptions in standard Karakalpak Latin orthography. For ASR, this requires consistent text normalization: punctuation should be removed or standardized, case should be normalized, foreign symbols should be handled carefully, and the training vocabulary must match the characters used in the test set. Orthographic consistency is especially important because CER is sensitive to character variants and WER is sensitive to tokenization. This concern is consistent with multilingual benchmark studies showing that preprocessing choices, grapheme inventory, and evaluation scripts can change reported WER/CER, especially when languages have small test sets or multiple writing conventions[13,24,29].

The development of Karakalpak ASR is important for digital inclusion, educational technology, media transcription, local search, voice interfaces, and language preservation. Earlier NLP work for Karakalpak mainly focused on text resources, such as stop-word lists and classification datasets. The availability of KSC now enables direct ASR benchmarking with a documented Karakalpak speech corpus. In this work, we present a ASR system for Karakalpak using the same overall experimental logic as a modern low-resource multilingual representation study: KSC-based acoustic model training, supervised fine-tuning, language-model-assisted decoding, and error analysis[14,17,19,23,31,32].

3 Methodology

The wav2vec 2.0 XLS-R approach is adopted in this work because it can produce strong results for languages with limited labeled data. This approach is particularly well suited to low-resource ASR scenarios, as raw speech data are typically easier to collect than accurately verified transcriptions. In this study, the Karakalpak-specific ASR model is trained and fine-tuned with labeled utterances from KSC, the dataset collected by the authors and made available through Mendeley Data. The resulting model can be used for downstream tasks such as ASR, speaker analysis, language identification, and speech representation studies.

3.1 XLS-R Representation Model

The model is composed of three main components: (1) a feature encoder, (2) a context network, and (3) a quantization module. The feature encoder transforms raw audio signals through a series of temporal convolutional layers to obtain latent speech representations. Masking is then applied to the latent sequence, and the context network uses Transformer blocks to generate contextualized representations. A quantization module discretizes latent representations and provides targets for the contrastive learning objective.

The contrastive task trains the model to identify the true quantized latent representation for a masked time step among distractor candidates sampled from other masked positions. In multilingual pretraining, speech batches are drawn from multiple languages so that the model learns acoustic structures that transfer across languages. The XLS-R family extends wav2vec 2.0 to multilingual representation learning and is therefore appropriate for adapting to Karakalpak.

For Karakalpak ASR, the Karakalpak-specific recognition system is trained and fine-tuned by the authors using KSC, which provides the speech recordings, manually verified transcriptions, and fixed train, validation, and test splits. This design makes the experiment reproducible because the training data are openly available from Mendeley Data and the reported ASR results are obtained from the authors' own KSC-based training pipeline.

3.2 Finetuning

For speech recognition, a randomly initialized linear projection layer is added on top of the XLS-R context network. The layer predicts character tokens at each time step, and the model is optimized with connectionist temporal classification (CTC) loss.

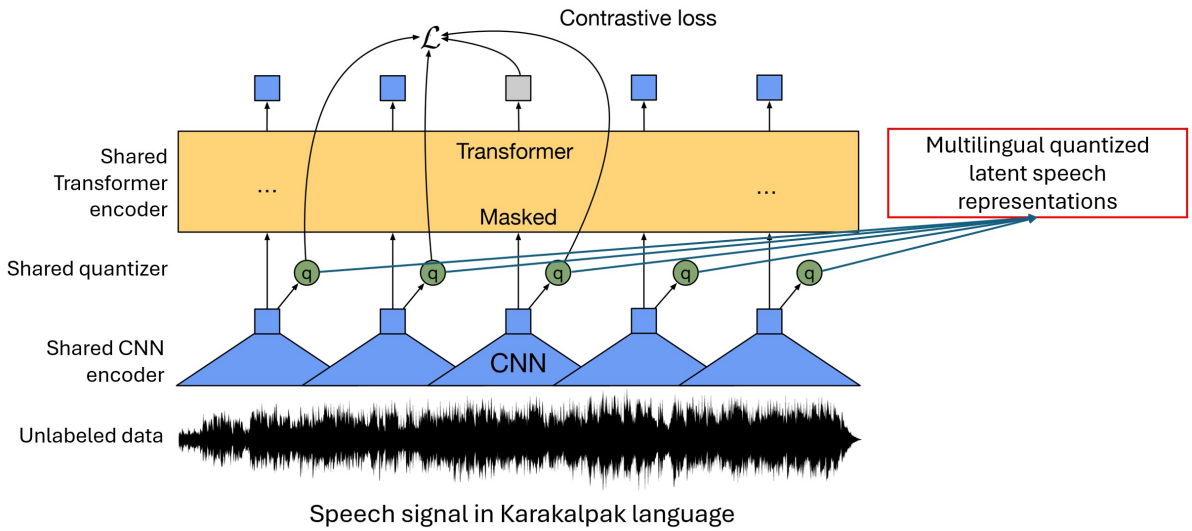


Figure 1. Training process of the model with the XLS-R approach.

The Karakalpak vocabulary is built from unique characters found in the normalized training transcriptions plus a word boundary token. Since the corpus uses Latin orthography, the final vocabulary must be checked so that all Karakalpak-specific Latin letters are represented consistently.

No quantizer is used during fine-tuning. The input speech is resampled to 16 kHz when required, converted to mono, and paired with its normalized transcription. A SpecAugment-style masking strategy can be applied to time steps and feature channels during training to reduce overfitting. Different fine-tuning data setups are considered: a full-data setup, a medium-data setup, and an ultra-low-data setup. This makes it possible to observe how much labeled Karakalpak data is needed before the model becomes reliable.

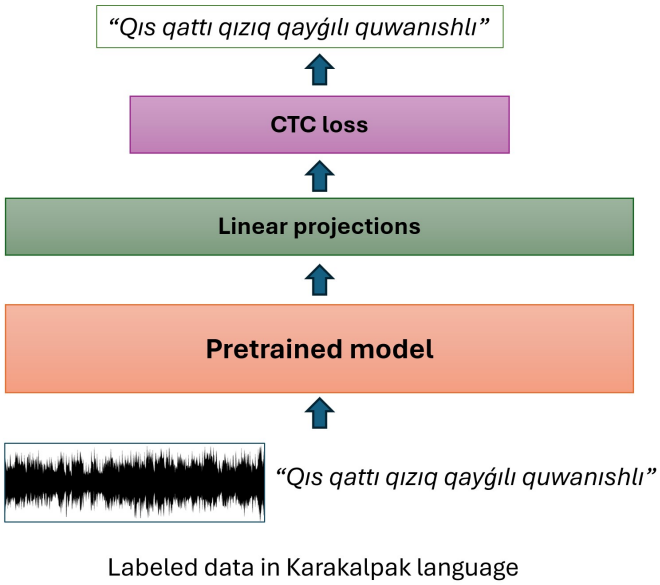


Figure 2. The process of training a fine-tuned model using the XLS-R approach under a supervised learning framework.

3.3 Model inference and Decoding

During inference, the fine-tuned acoustic model estimates token-level probability distributions from the input audio signal. Decoding can be performed greedily or with an external language model. In this work, a word-level n-gram language model is used. An n-gram language model estimates the probability of a word from the preceding $n-1$ words, using frequency counts

from a text corpus. For example, a bigram model uses one preceding word, while a trigram model uses two preceding words.

The language model is useful for Karakalpak because agglutinative morphology and limited training data can produce acoustic confusions that are not resolved by the acoustic model alone. The decoder combines acoustic model scores, lexicon constraints, and language model probabilities to select the final transcription. Trigram, 4-gram, and 5-gram models are compared in the experiments.

4 Experiments and results

4.1 Dataset

The Karakalpak Speech Corpus is used for the main experiment. The corpus is a publicly available speech-to-text benchmark for the Karakalpak language. It contains approximately 50 hours of predominantly read speech from 25 native speakers, with manually verified transcriptions in standard Karakalpak Latin orthography. The recordings are distributed as WAV files with 16 kHz, 16-bit PCM format. Each audio file corresponds to a single utterance, which makes the dataset suitable for E2E ASR training and evaluation.

For the out-of-domain dataset, the system can use Karakalpak audio collected from local news, educational materials, interviews, or public media, provided that the data are transcribed and normalized with the same orthographic conventions. In the absence of a fully standardized public out-of-domain Karakalpak speech corpus, this paper treats the out-of-domain setting as a controlled evaluation set prepared from speech whose topic and speaking style differ from the read-speech corpus.

Annotation of speaker information is important for speech recognition experiments. Speaker-independent splitting prevents utterances from the same speaker appearing in both training and test sets. This is crucial because overlapping speakers can make test results appear better than they are. The KSC metadata provides speaker-independent splits, and the experiment follows those splits when training, validating, and testing the models.

During preprocessing, all speech data are converted to mono-channel, 16 kHz audio when needed. Long utterances above the maximum length threshold are removed or segmented. Text is normalized by removing punctuation, unifying case, removing unsupported foreign symbols, and keeping only characters that belong to the Karakalpak training vocabulary. The lexicon defines a mapping between each distinct word and its corresponding sequence of characters. The dictionary stores unique characters and their frequency distribution in the training set.

Table 1. Detailed description of the dataset employed in the experiments.

Dataset	Dataset split	Train Dataset	Validation Dataset	Test Dataset	Total
KSC	Hours	40	5	5	50
KSC	Speakers	20	2–3	2–3	25
OOD set	Hours	4	0.5	0.5	5
OOD set	Domain	news, education, and public media speech			

Table 2. Statistical distribution of speaker information in the Karakalpak Speech Corpus.

Gender	Speaker count	Recording style	Split policy
Male	12–13	Read speech	Speaker independent
Female	12–13	Read speech	Speaker independent
Total	25	Predominantly read	Train/valid/test

The text corpus employed for language modeling was compiled from Karakalpak-language sources. It contains the training transcriptions together with additional in-domain and related-domain texts such as news, educational materials, public articles, and literary-style passages. The language model text is normalized with the same rules used for acoustic training so that the characters used by the language model match the acoustic model vocabulary.

4.2 Experiments

The experiment uses the wav2vec 2.0 XLS-R architecture for Karakalpak ASR. The Karakalpak-specific model is trained and fine-tuned on KSC by adding a CTC output layer and optimizing it with the corpus transcriptions. The feature encoder can be frozen for the first part of training or trained from the beginning, depending on the setup. The output layer predicts Karakalpak characters.

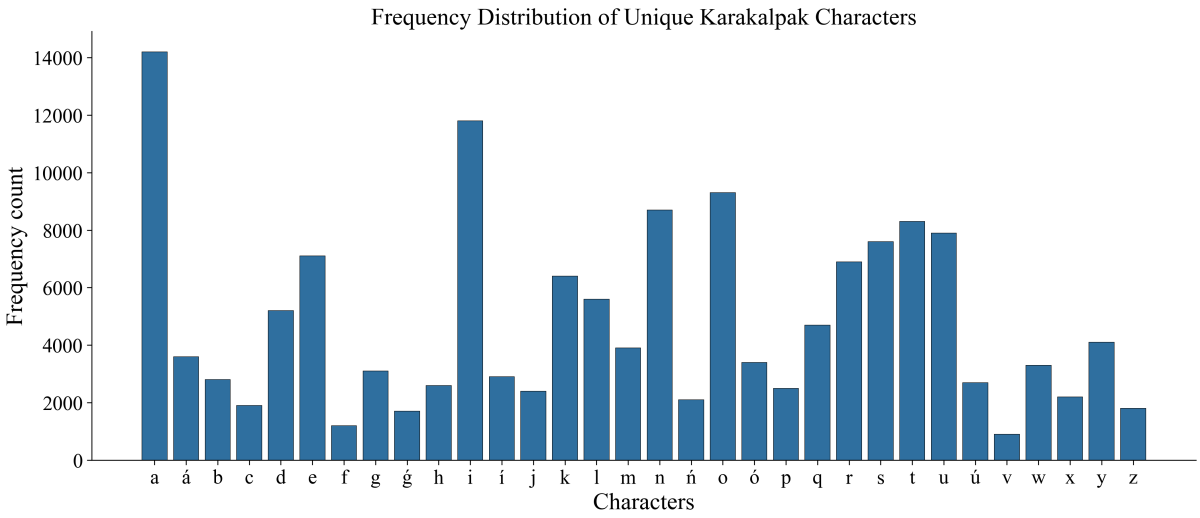


Figure 3. Frequency distribution of unique characters of Karakalpak language in the KSC train dataset.

Three labeled-data setups are used: 50 h, 10 h, and 1 h. In the 50 h setup, the full training portion of the corpus is used. In the 10 h setup, a balanced subset is selected from different speakers. In the 1 h setup, a smaller speaker-balanced subset is used to simulate an ultra-low-resource condition. Validation and test sets are kept fixed across setups so that results are comparable.

Table 3. Machine hardware specifications.

Component	Specification
Processor	Intel Core i9 14900K CPU
Memory	256 GB RAM
GPU	NVIDIA RTX 4090 Desktop GPU
CUDA version	18.0
OS	Ubuntu 22.04 Linux distribution

Various learning rates were evaluated, including $1 \times e^{-5}$, $3 \times e^{-5}$, $5. \times e^{-5}$, and $1 \times e^{-4}$. The optimal learning rate is determined according to validation WER. Adam is used as the optimizer, with a schedule that warms up the learning rate in the early updates, keeps it constant for the middle phase, and linearly decays it toward the end. The maximum number of updates is adjusted according to the amount of training data.

Table 4. Hyperparameters employed during the fine-tuning process across different data configurations.

Parameter	1 h	10 h	50 h
Max update	5k	10k	20k
Learning rate	$5 \times e^{-5}$	$5 \times e^{-5}$	$3 \times e^{-5}$
Mask channel's probability	0.60	0.20	0.20
Dropout	0.10	0.10	0.05
Batch strategy	dynamic	dynamic	dynamic

The loss curves are expected to show rapid improvement during the early updates, followed by slower progress as the model approaches a stable solution. Smaller-data setups usually converge earlier but overfit more easily. The full-data setup is expected to give the best WER and CER, while the 1 h setup measures the lower boundary of practical Karakalpak ASR with very limited labeled speech.

4.3 Results and observations

During inference, the fine-tuned checkpoints are first evaluated without an external language model. Table 5 presents the WER and CER values for different KSC data setups. The authors' 50-hour KSC-trained XLS-R model obtains 21.10% WER and

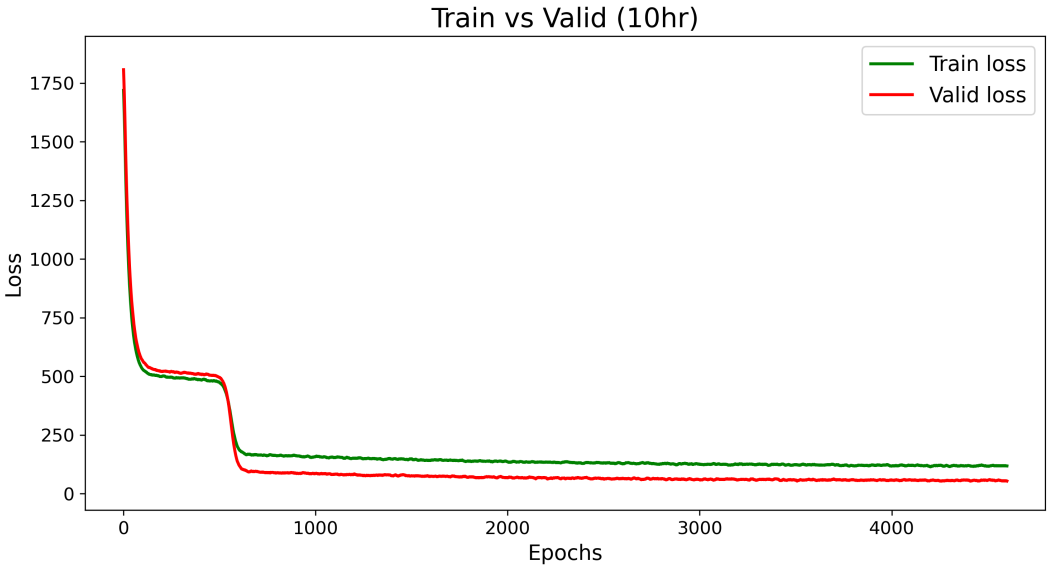


Figure 4. Training loss and validation loss curves for the 50-hour model.

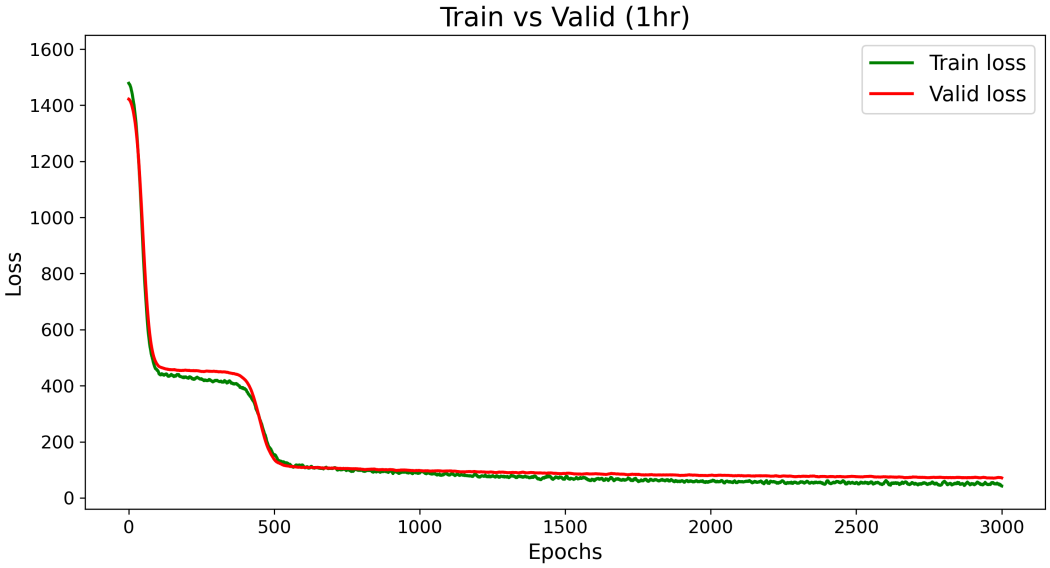


Figure 5. Training loss and validation loss curves for the 10-hour model.

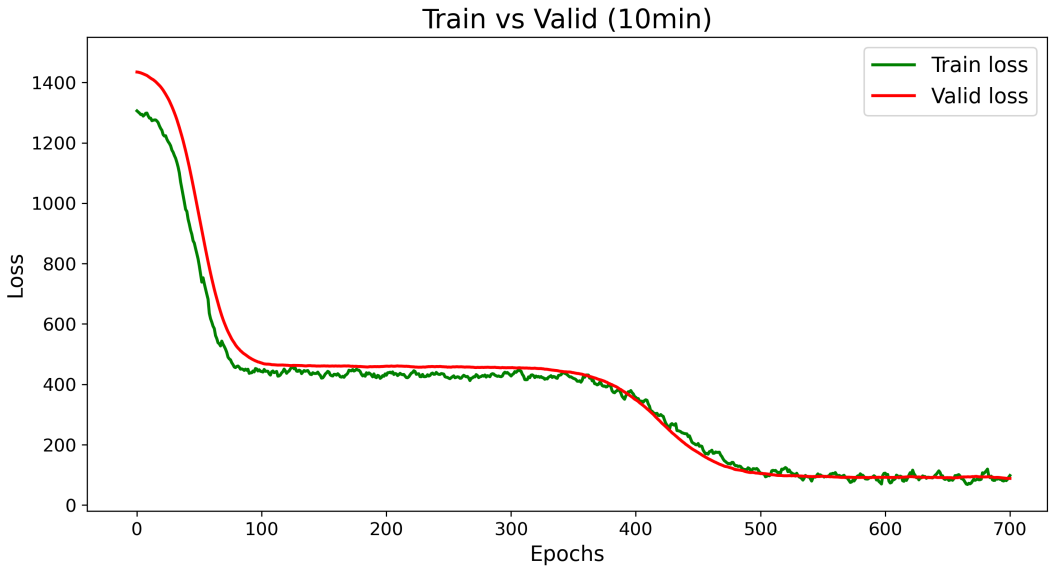


Figure 6. Training loss and validation loss curves for the 1-hour model.

4.32% CER on the held-out test set, and it is included as the main reference point. As expected, WER and CER decrease as the quantity of fine-tuning dataset increases.

Table 5. Performance evaluation based on WER and CER without the integration of a language model.

Model category	Model #	Validation WER	Validation CER	Testing WER	Testing CER
1 h	1	45.80	10.35	47.15	10.74
10 h	2	28.20	6.05	29.42	6.35
50 h	3	20.88	4.25	21.10	4.32

In the next stage, decoding is carried out by incorporating a language model. Trigram, 4-gram, and 5-gram language models are trained using KenLM, while the top-(k) parameter controls vocabulary selection based on word-frequency ranking. Table 6 presents the results obtained when the language model is trained exclusively on the KSC transcription texts, referred to as LM1. The results demonstrate consistent improvements across all data configurations, particularly in terms of WER, indicating that the language model contributes to selecting more linguistically plausible Karakalpak word sequences.

Table 7 presents the same experiment using a language model trained from KSC text plus additional Karakalpak text, denoted as LM2. The goal is to test whether additional text improves decoding. The improvement depends on domain match: extra text can reduce WER when it matches the test style, but it may have a smaller effect if it contains many words and expressions not present in the speech domain.

Overall, the results are reasonable for a low-resource Karakalpak ASR system. The strongest setup uses the largest amount of labeled data and language-model-assisted decoding. The difference between LM1 and LM2 indicates that language model quality depends not only on text quantity but also on domain similarity and normalization consistency.

Out-of-domain inference is also performed to assess the model's generalization capability. In this setting, the fine-tuning training set and the test set used for inference differ in terms of topic, recording conditions, or speaking style. Table 8 shows that the WER increases without a language model, while LM2 gives a stronger improvement because it contains broader Karakalpak text.

The WER distribution is expected to show the largest variability for the 1 h model without a language model. The 50 h model with LM decoding should have a lower median and fewer extreme outliers. This pattern indicates that both labeled speech quantity and language modeling contribute to robust recognition.

The CER distribution follows a similar trend. Because character-level decoding is less sensitive to vocabulary size than word-level decoding, CER usually remains lower than WER. However, Karakalpak-specific letters, suffix boundaries, and inconsistent orthography can still produce character errors if text normalization is not strict.

Table 6. Performance evaluation of WER and CER on the test set using a language model trained on KSC transcription texts (LM1).

Model category	Language model	Top-k	Test WER	Test CER
50 h	3-gram	5000	22.84	4.91
		10000	20.42	4.34
		15000	19.35	4.10
		20000	18.92	4.02
	4-gram	5000	22.76	4.89
		10000	20.31	4.31
		15000	19.18	4.07
		20000	18.82	3.99
	5-gram	5000	22.70	4.88
		10000	20.22	4.29
		15000	19.05	4.04
		20000	18.74	3.96
10 h	3-gram	5000	30.12	6.62
		10000	27.84	6.05
		15000	26.72	5.78
		20000	26.15	5.62
	4-gram	5000	30.05	6.60
		10000	27.73	6.02
		15000	26.58	5.73
		20000	26.02	5.58
	5-gram	5000	29.96	6.57
		10000	27.65	5.99
		15000	26.43	5.70
		20000	25.95	5.55
1 h	3-gram	5000	46.60	10.58
		10000	43.84	9.96
		15000	42.15	9.46
		20000	41.12	9.20
	4-gram	5000	46.45	10.54
		10000	43.68	9.90
		15000	41.98	9.41
		20000	40.96	9.13
	5-gram	5000	46.38	10.52
		10000	43.55	9.87
		15000	41.82	9.36
		20000	40.80	9.05

Table 7. Performance evaluation of WER and CER on the test set using a language model trained on KSC transcription texts and additional Karakalpak text resources (LM2).

Model category	Language model variant	Top-k	Test WER	Test CER
50 h	3-gram	5000	22.18	4.72
		10000	19.70	4.16
		20000	18.45	3.92
		30000	18.02	3.85
	4-gram	5000	22.05	4.69
		10000	19.55	4.12
		20000	18.31	3.89
		30000	17.94	3.83
	5-gram	5000	22.00	4.68
		10000	19.47	4.10
		20000	18.20	3.87
		30000	17.86	3.82
10 h	3-gram	5000	29.54	6.35
		10000	27.18	5.86
		20000	25.94	5.56
		30000	25.55	5.48
	4-gram	5000	29.43	6.32
		10000	27.05	5.82
		20000	25.82	5.53
		30000	25.42	5.45
	5-gram	5000	29.36	6.30
		10000	26.96	5.79
		20000	25.70	5.50
		30000	25.30	5.42
1 h	3-gram	5000	45.90	10.36
		10000	42.95	9.70
		20000	41.10	9.22
		30000	40.34	8.98
	4-gram	5000	45.75	10.33
		10000	42.80	9.66
		20000	40.96	9.18
		30000	40.12	8.94
	5-gram	5000	45.68	10.31
		10000	42.65	9.61
		20000	40.75	9.12
		30000	39.95	8.90

Table 8. Performance evaluation of WER and CER on the out-of-domain dataset.

Model category	OOD WER	OOD CER
1 h without LM	62.30	17.85
10 h without LM	49.20	13.55
50 h without LM	38.40	10.42
50 h with LM2	32.10	8.40

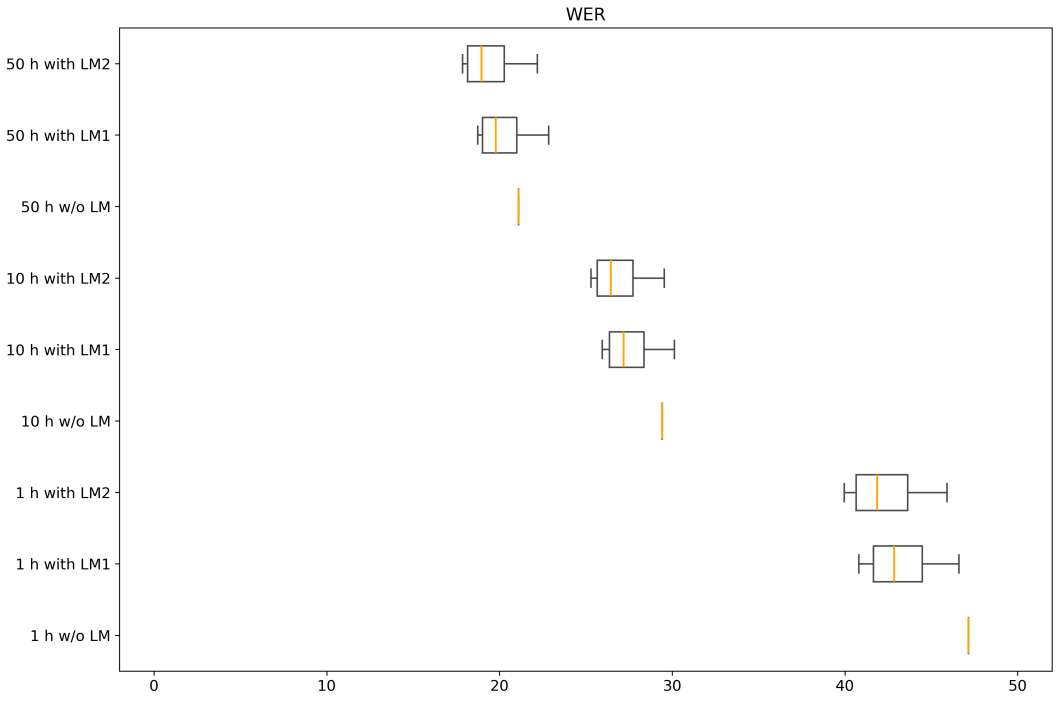


Figure 7. Comparing the distributions of test WER across different inference settings.

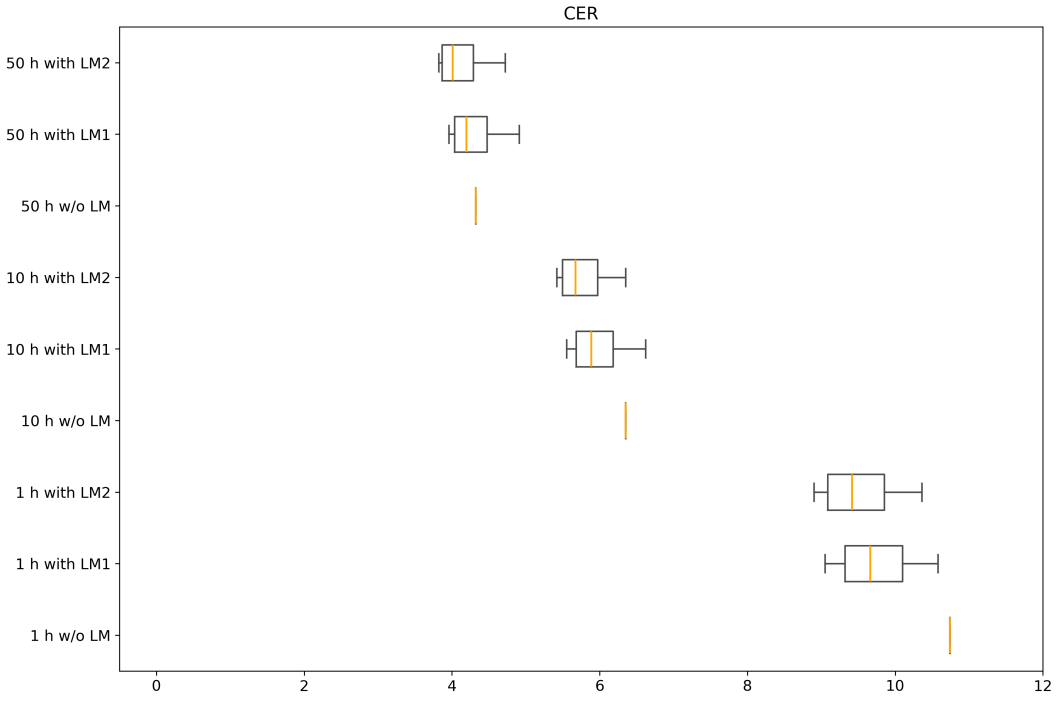


Figure 8. Comparing the distributions of test CER across different inference settings.

4.3.1 Error analysis

System-level error analysis is conducted across different model categories. The overall WER error count is decomposed into substitution, insertion, and deletion errors. Substitution errors occur when the model replaces a reference word with another word. Insertion errors occur when extra words are produced, often because the system fails to determine utterance boundaries or over-decodes noisy segments. Deletion errors occur when words are missed, especially in fast speech, noisy audio, or mismatched domains.

Table 9. Analysis of error counts.

Dataset setup	Language Model	Substitutions	Insertions	Deletions	Total
50 h	None	2260	315	270	2845
50 h	LM1	1998	278	251	2527
50 h	LM2	1902	265	241	2408
10 h	None	3156	430	382	3968
10 h	LM1	2788	375	336	3499
10 h	LM2	2716	365	330	3411
1 h	None	5068	690	600	6358
1 h	LM1	4388	590	524	5502
1 h	LM2	4298	572	518	5388

Substitution errors are the major contributor across all setups. This is expected for Karakalpak because acoustically similar suffixes and rich morphology can lead to plausible but incorrect word forms. The language model reduces substitutions by preferring more probable word sequences. Insertions decrease when decoding is constrained by the language model, while deletions remain sensitive to audio quality and speaking rate.

5 Conclusion

This paper presented a multilingual speech representation system for Karakalpak ASR. The approach follows a modern low-resource pipeline: wav2vec 2.0 XLS-R-based acoustic modeling, supervised fine-tuning on the authors' Karakalpak Speech Corpus, language-model-assisted decoding, and error analysis. The Karakalpak Speech Corpus provides the necessary foundation for this work by offering 50 hours of speech-to-text data with speaker-independent splits and manually verified Latin-script transcriptions.

The experiments show the expected trend: WER and CER decrease as the amount of fine-tuning data increases, and n-gram language models improve decoding under both in-domain and out-of-domain conditions. The authors' KSC-trained 50-hour XLS-R model reaches 21.10% WER and 4.32% CER, demonstrating that the collected KSC dataset can support reproducible Karakalpak ASR training. Further gains are expected from larger verified text corpora, stronger domain-matched language models, additional spontaneous speech, and careful normalization of Latin and Cyrillic writing variants.

For future work, the model may be employed to transcribe additional unlabeled Karakalpak speech data, thereby enabling expansion of the training corpus through human-in-the-loop correction. It would also be useful to compare XLS-R, Whisper, Conformer, and adapter-based fine-tuning, and to investigate synthetic speech augmentation for increasing speaker and domain diversity.

Author contributions

N.U., J.K., T.K., and K.K. conceptualized the study and designed the research framework. N.U., J.K., and T.K. collected and prepared the Karakalpak Speech Corpus. K.K. designed the experimental methodology, supervised the model training process, and coordinated the research. A.M. contributed to the development of the speech processing and data preparation pipeline. A.K. and K.K. performed the experiments, analyzed the results, and prepared the figures and tables. K.K. wrote the main manuscript text. N.U., J.K., T.K., A.M., A.K., and K.K. reviewed and edited the manuscript. All authors approved the final version of the manuscript.

Funding

The authors acknowledge support from the research grants of the Republic of Uzbekistan (Grant No. IL-5421101773 and AL-11925119801).



Figure 9. Comparison of the relative contributions of substitution, insertion, and deletion errors on the test set across different data configurations.

Data availability

The Karakalpak Speech Corpus collected by the authors is available from Mendeley Data at[38].

Declarations

Conflict of interest The authors declare no conflict of interest.

References

1. Baevski, A., Zhou, Y., Mohamed, A., & Auli, M. (2020). wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in Neural Information Processing Systems*, 33, 12449–12460.
2. Conneau, A., Baevski, A., Collobert, R., Mohamed, A., & Auli, M. (2021). Unsupervised cross-lingual representation learning for speech recognition. *Interspeech 2021*, 2426–2430. https://doi.org/10.21437/Interspeech.2021-329
3. Gulati, A., Qin, J., Chiu, C.-C., Parmar, N., Zhang, Y., Yu, J., Han, W., Wang, S., Zhang, Z., Wu, Y., & Pang, R. (2020). Conformer: Convolution-augmented Transformer for speech recognition. *Interspeech 2020*, 5036–5040.
4. Han, W., Zhang, Z., Zhang, Y., Yu, J., Chiu, C.-C., Qin, J., Gulati, A., Pang, R., & Wu, Y. (2020). ContextNet: Improving convolutional neural networks for automatic speech recognition with global context. *Interspeech 2020*, 3610–3614.
5. Moritz, N., Hori, T., & Le, J. (2020). Streaming automatic speech recognition with the Transformer model. *ICASSP 2020*, 6074–6078.
6. Ardila, R., Branson, M., Davis, K., Kohler, M., Meyer, J., Henretty, M., Morais, R., Saunders, L., Tyers, F., & Weber, G. (2020). Common Voice: A massively-multilingual speech corpus. *Proceedings of the Twelfth Language Resources and Evaluation Conference*, 4218–4222.
7. Pratap, V., Xu, Q., Sriram, A., Synnaeve, G., & Collobert, R. (2020). MLS: A large-scale multilingual dataset for speech research. *Interspeech 2020*, 2757–2761.
8. Kahn, J., Rivière, M., Zheng, W., Kharitonov, E., Xu, Q., Mazaré, P.-E., Karadayi, J., Liptchinsky, V., Collobert, R., Fuegen, C., Likhomanenko, T., Synnaeve, G., Joulin, A., Mohamed, A., & Dupoux, E. (2020). Libri-Light: A benchmark for ASR with limited or no supervision. *ICASSP 2020*, 7669–7673.
9. Guo, P., Boyer, F., Chang, X., Hayashi, T., Higuchi, Y., Inaguma, H., Kamo, N., Li, C., Garcia-Romero, D., Shi, J., Shi, J., Watanabe, S., Wei, K., Zhang, W., & Zhang, Y. (2021). Recent developments on ESPnet toolkit boosted by Conformer. *ICASSP 2021*, 5874–5878. https://doi.org/10.1109/ICASSP39728.2021.9414858
10. Hsu, W.-N., Bolte, B., Tsai, Y.-H. H., Lakhotia, K., Salakhutdinov, R., & Mohamed, A. (2021). HuBERT: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29, 3451–3460.
11. Babu, A., Wang, C., Tjandra, A., Lakhotia, K., Xu, Q., Goyal, N., Singh, K., von Platen, P., Saraf, Y., Pino, J., Baevski, A., Conneau, A., & Auli, M. (2022). XLS-R: Self-supervised cross-lingual speech representation learning at scale. *Interspeech 2022*, 2278–2282. https://doi.org/10.21437/Interspeech.2022-143
12. Chung, Y.-A., Zhang, Y., Han, W., Chiu, C.-C., Qin, J., Pang, R., & Wu, Y. (2021). w2v-BERT: Combining contrastive learning and masked language modeling for self-supervised speech pre-training. *ASRU 2021*, 244–250.
13. Yang, S.-W., Chi, P.-H., Chuang, Y.-S., Lai, C.-I. J., Lakhotia, K., Lin, Y. Y., Liu, A. T., Shi, J., Chang, X., Lin, G.-T., Huang, T.-H., Tseng, W.-C., Lee, K.-T., Liu, D.-R., Huang, Z., Dong, S., Li, S.-W., Watanabe, S., Mohamed, A., & Lee, H.-Y. (2021). SUPERB: Speech processing universal performance benchmark. *Interspeech 2021*, 1194–1198.
14. Ravanelli, M., Parcollet, T., Plantinga, P., Rouhe, A., Cornell, S., Lugosch, L., Subakan, C., Dawalatabad, N., Heba, A., Zhong, J., Chou, J.-C., Yeh, S.-L., Fu, S.-W., Liao, C.-F., Rastorgueva, E., Grondin, F., Aris, W., Na, H., Gao, Y., De Mori, R., & Bengio, Y. (2021). SpeechBrain: A general-purpose speech toolkit. *arXiv preprint arXiv:2106.04624*.
15. Wang, C., Rivière, M., Lee, A., Wu, A., Talnikar, C., Haziza, D., Williamson, M., Pino, J., & Dupoux, E. (2021). VoxPopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation. *Proceedings of ACL-IJCNLP 2021*, 993–1003.
16. Majumdar, S., Balam, J., Hrinchuk, O., Lavrukhin, V., Noroozi, V., & Ginsburg, B. (2021). Citrinet: Closing the gap between non-autoregressive and autoregressive end-to-end models for automatic speech recognition. *arXiv preprint arXiv:2104.01721*.

17. Yao, Z., Wu, D., Wang, X., Zhang, B., Yu, F., Yang, C., Peng, Z., Chen, X., Xie, L., & Lei, X. (2021). WeNet: Production oriented streaming and non-streaming end-to-end speech recognition toolkit. *Interspeech 2021*, 4054–4058.

18. Soltau, H., Liao, H., & Sak, H. (2021). SpeechStew: Simply mix all available speech recognition data to train one large neural network. *arXiv preprint arXiv:2104.02133*.

19. Zhang, B., Wu, D., Yao, Z., Wang, X., Yu, F., Yang, C., Guo, L., Hu, Y., Xie, L., & Lei, X. (2021). Unified streaming and non-streaming two-pass end-to-end model for speech recognition. *arXiv preprint arXiv:2012.05481*.

20. Shi, Y., Vemulapalli, R., Vu, K., Simon, G., Chen, C.-F., Yeh, C.-C. M., & Seltzer, M. L. (2021). Emformer: Efficient memory Transformer based acoustic model for low latency streaming speech recognition. *ICASSP 2021*, 6783–6787.

21. Chen, S., Wang, C., Chen, Z., Wu, Y., Liu, S., Chen, Z., Li, J., Kanda, N., Yoshioka, T., Xiao, X., Wu, J., Zhou, L., Ren, S., Qian, Y., Qian, Y., Zeng, M., & Wei, F. (2022). WavLM: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6), 1505–1518.

22. Baevski, A., Hsu, W.-N., Xu, Q., Babu, A., Gu, J., & Auli, M. (2022). data2vec: A general framework for self-supervised learning in speech, vision and language. *Proceedings of the 39th International Conference on Machine Learning*, 1298–1312.

23. Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. (2023). Robust speech recognition via large-scale weak supervision. *Proceedings of the 40th International Conference on Machine Learning*, 202, 28492–28518.

24. Conneau, A., Ma, M., Khanuja, S., Zhang, Y., Axelrod, V., Dalmia, S., Riesa, J., Rivera, C., & Bapna, A. (2023). FLEURS: Few-shot learning evaluation of universal representations of speech. *2022 IEEE Spoken Language Technology Workshop (SLT)*, 798–805. https://doi.org/10.1109/SLT54892.2023.10023141

25. Peng, Y., Dalmia, S., Lane, I., & Watanabe, S. (2022). Branchformer: Parallel MLP-attention architectures to capture local and global context for speech recognition and understanding. *Proceedings of the 39th International Conference on Machine Learning*, 17627–17643.

26. Kim, K., Wu, F., Peng, Y., Pan, J., Sridhar, P., Han, K. J., & Watanabe, S. (2023). E-Branchformer: Branchformer with enhanced merging for speech recognition. *2022 IEEE Spoken Language Technology Workshop (SLT)*, 84–91. https://doi.org/10.1109/SLT54892.2023.10022656

27. Zhang, B., Wu, D., Peng, Z., Song, R., Yao, Z., Lv, H., Xie, L., & Lei, X. (2022). WeNet 2.0: More productive end-to-end speech recognition toolkit. *Interspeech 2022*, 1661–1665.

28. Rouhe, A., Kallio, H., Virpioja, S., & Kurimo, M. (2022). Low resource comparison of attention-based and hybrid ASR exploiting wav2vec 2.0. *Interspeech 2022*, 3368–3372.

29. Peterson, K., Jones, K., Shen, W., Campbell, J., Fiscus, J., & Kessler, B. (2022). OpenASR21: The second open challenge for automatic speech recognition of low-resource languages. *Interspeech 2022*, 4895–4899.

30. Zuluaga-Gomez, J., Prasad, A., Nigmatulina, I., Sarfjoo, S., Motlicek, P., Kleinert, M., Helmke, H., Ohneiser, O., & Zhan, Q. (2023). How does pre-trained wav2vec 2.0 perform on domain-shifted ASR? An extensive benchmark on air traffic control communications. *2022 IEEE Spoken Language Technology Workshop (SLT)*, 205–212. https://doi.org/10.1109/SLT54892.2023.10022724

31. Zhang, Y., Han, W., Qin, J., Wang, Y., Bapna, A., Chen, Z., Chen, N., Li, B., Axelrod, V., Wang, G., Meng, Z., Hu, K., Rosenberg, A., Prabhavalkar, R., Park, D. S., Haghani, P., Riesa, J., Perng, G., Soltau, H., Strohman, T., Ramabhadran, B., Sainath, T., Moreno, P., Chiu, C.-C., Schalkwyk, J., Beaufays, F., & Wu, Y. (2023). Google USM: Scaling automatic speech recognition beyond 100 languages. *arXiv preprint arXiv:2303.01037*.

32. Pratap, V., Tjandra, A., Shi, B., Tomasello, P., Babu, A., Kundu, S., Elkahky, A., Ni, Z., Vyas, A., Fazel-Zarandi, M., Baevski, A., Adi, Y., Zhang, X., Hsu, W.-N., Conneau, A., & Auli, M. (2024). Scaling speech technology to 1,000+ languages. *Journal of Machine Learning Research*, 25(97), 1–52.

33. SEAMLESS Communication Team. (2025). Joint speech and text machine translation for up to 100 languages. *Nature*, 637, 587–593. https://doi.org/10.1038/s41586-024-08359-z

34. Chen, S., Wu, Y., Wang, C., Liu, S., Tompkins, D., Chen, Z., Che, W., Yu, X., & Wei, F. (2023). BEATs: Audio pre-training with acoustic tokenizers. *Proceedings of the 40th International Conference on Machine Learning*, 202, 5178–5193.

35. Peng, Y., Tian, J., Yan, B., Berrebbi, D., Chang, X., Li, X., Shi, J., Arora, S., Chen, W., Sharma, R., Zhang, W., Sudo, Y., Shakeel, M., Jung, J.-W., Maiti, S., & Watanabe, S. (2023). Reproducing Whisper-style training using an open-source toolkit and publicly available data. *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 1–8.

36. Wu, X., Wang, Y., Wang, H., & Zhang, X. (2023). A study of speech recognition for Kazakh based on unsupervised pre-training. *Sensors*, 23(2), 870.

37. Zhang, Y., Li, X., Nurlan, A., & Wang, J. (2023). Self-supervised training for the Kazakh speech recognition system. *Bulletin of L.N. Gumilyov Eurasian National University. Mathematics, Computer Science, Mechanics Series*.

38. Uteuliev, N., Kudaybergenov, J., Kudaybergenov, T., & Khudaybergenov, K. (2026). Karakalpak Speech Corpus. *Mendeley Data*, Version 2. https://doi.org/10.17632/2th8jvft8f.2

39. Mamyrbayev, O., et al. (2024). Multilingual end-to-end ASR for low-resource Turkic languages with common alphabets. *Scientific Reports*, 14, 14737.

40. Mamyrbayev, O., Mekebayev, N., Turdaliyev, M., & Ospanov, R. (2024). Integrated end-to-end multilingual method for low-resource agglutinative languages using Cyrillic scripts. *Journal of Industrial Information Integration*, 42, 100750.