Enhance Contextual Learning in ASR for Endangered Low-resource Languages

Zhaolin Li, Jan Niehues
Karlsruhe Institute of Technology, Germany
{zhaolin.li, jan.niehues}@kit.edu

Abstract

Automatic Speech Recognition (ASR) facilitates documenting endangered low-resource languages. While recent advances in acoustic modelling have been substantial, contextual learning remains underexplored. This study investigates the main factors that influence the integration of knowledge from language models (LMs) into state-of-the-art ASR models for endangered low-resource languages. Through experiments on five diverse low-resource languages, we find: 1) Fine-grained tokenization effectively improves ASR performance by addressing the prevalent unknown words and improving data usage efficiency; 2) The integration of transformer-based LMs into ASR systems surpasses that of N-gram LMs only in one language, even though they consistently achieve better results in language modelling tasks. 3) ASR performance is highly sensitive to language-specific optimization, as shown by a 43% performance degradation in one language due to parameter transfer across languages. We open-source our scripts to support further research and applications [1].

1 Introduction

The threat of language endangerment continues to grow due to various external pressures, prompting linguists to actively document vulnerable languages. However, manual documentation processes are often impractical and time-intensive. Automatic Speech Recognition (ASR) models offer valuable support for language documentation, yet their effectiveness is hindered by the limited availability of supervised data.

Recent advancements indicate that multilingual self-supervised learning holds promise for developing ASR systems tailored to endangered low-resource languages(Mihajlik et al., 2023; Li et al., 2024; Taguchi et al., 2024; Mainzinger and Levow, 2024; Taguchi and Chiang, 2024). Among these approaches, fine-tuning the pre-trained Wav2Vec2 models (Conneau et al., 2020) with Connectionist Temporal Classification (CTC) loss (Graves et al., 2006) has emerged as a popular and effective strategy. Compared to other pre-trained ASR models, such as Whisper (Radford et al., 2023), this approach often achieves superior performance, particularly in reducing character-level errors (Le Ferrand et al., 2024; He et al., 2024). This advantage can be attributed to its smaller parameter set and the extensive pre-training data, making it especially effective for low-resource settings.

Despite its strengths in acoustic modelling, this approach lacks contextual learning capabilities due to the conditional independence assumption inherent in CTC (Graves et al., 2006; Lu and Chen, 2023; Higuchi et al., 2022). To address this, previous research has integrated ASR models with language models (LMs) at the word level (Conneau et al., 2020; San et al., 2023; Liu et al., 2024; He et al., 2024; Pratap et al., 2024; Arisaputra et al., 2024). However, word-level integration struggles with the high prevalence of unknown words in low-resource settings, where limited text data further impedes performance.

Additionally, prior studies have predominantly employed statistical N-gram LMs for integration. However, transformer-based LMs have demonstrated superior contextual learning capabilities compared to N-gram models for high-resource languages. While few studies have explored combining transformer-based LMs with Wav2Vec2 and CTC fine-tuning (Conneau et al., 2020), these investigations have focused on high-resource languages, leaving their potential for low-resource languages unexplored. Differences in data availability and linguistic complexity underscore the need for further investigation.

To fill these gaps, we explore LM integration for five low-resource languages from diverse language

[1] https://github.com/ZL-KA/LM-LR-ASR

Proceedings of the 1st Workshop on Language Models for Underserved Communities (LM4UC 2025), pages 1–7
May 4, 2025 ©2025 Association for Computational Linguistics

families, considering differences in data size and source. Our main contributions are:

1. Fine-grained tokenizations at subword and character levels generally improve performance, except for Khinalug, a language where minimal data availability imposes constraints.

2. The transformer-based method outperforms the N-gram approach only with one language, unlike high-resource languages where transformer models consistently excel (Conneau et al., 2020), highlighting challenges in low-resource settings.

3. Parameter optimization is highly language-specific, with parameter transferring from one language to another resulting in a significant performance gap from optimal outcomes.

2 Language Model in ASR

2.1 Language Model Integration

The popular ASR system for low-resource languages leverages self-supervised pre-training followed by CTC-based fine-tuning. Due to the independence assumption inherent in CTC, the ASR system incorporates LMs during decoding to enhance contextual learning[2]. Specifically, LM integration occurs during inference-only decoding in an auto-regressive manner [3]. In accordance with the CTC algorithm, the character-level acoustic representations accumulate based on the space separator. The corresponding sequence of characters is collapsed using the CTC algorithm, and the LM assigns scores to the resulting text. The total score is computed using Equation 1:

$$score = \log P(\text{text}) + \alpha * LM(\text{text}) + \beta \quad (1)$$

Here, $\log P(\text{text})$ represents the acoustic hidden representation, and $LM(\text{text})$ denotes the LM score. The parameters α and β control the contribution of the LM and adjust the length of the generated sequences, respectively. LM integration enables the CTC-based ASR model to perform beam search, where the candidate sequence with the highest score is returned as the final prediction.

2.2 Tokenization Granularity

Since CTC-based fine-tuning operates at the character level, current word-level integration overlooks the fine-grained knowledge provided by CTC, leaving room for potential improvement. Additionally, word-level LMs struggle to handle the prevalence of unknown words in low-resource languages, leading to performance degradation.

This work proposes integrating LMs at the subword and character levels. We encode the transcript with space markers ("_") to denote word boundaries. Tokenization-specific ASR models and LMs are built using corresponding encoded text, enabling the models to leverage encoded knowledge effectively. This encoding increases the frequency of sequence patterns, improving data utilization efficiency for LMs. Furthermore, unknown words are decomposed into recognizable subwords or characters, reducing their negative impact on performance.

The study also investigates the impact of transformer-based LMs on LM integration. The integration process is adapted by modifying the scoring function to accommodate the transformer-based approach. Similar to N-gram LMs, log probabilities are used as LM scores.

3 Experimental Setups

3.1 Datasets

To address the unique challenges of building ASR systems for low-resource languages, such as language complexity, limited corpus size, and sparse audio sources, this study conducts experiments on five linguistically diverse languages to explore their practical application in language documentation.: Khinalug (Li et al., 2024), Kichwa (Taguchi et al., 2024), Mboshi (Godard et al., 2018), Japhug (Guillaume et al., 2022), and Bemba (Sikasote et al., 2023). Four of the selected languages are recognized as endangered, while Bemba is included to examine the impact of collecting additional supervised data. Table 1 illustrates the occurrence of unknown words in the development and test splits, highlighting the potential risks of overlooking them when using word-level LMs.

3.2 Modelling

Acoustic Model: We utilize the state-of-the-art version of Wav2Vec2 model mms-300 [4]. Pre-trained

[2] https://huggingface.co/blog/wav2vec2-with-ngram

[3] https://github.com/kensho-technologies/pyctcdecode/tree/main

[4] https://huggingface.co/facebook/mms-300m

Language	ISO code	Language Family	Audio source	Train (h)	Dev+Test (h)	Unknown words
Khinalug	kjj	Northeast Caucasian	Spontaneous	2.14	0.49	25.12%
Kichwa	que	Quechuan	Radio	3.05	0.77	27.28%
Mboshi	mdw	Bantu ZoneC	Reading	3.93	0.53	16.57%
Japhug	jya	Sino-Tibetan	Spontaneous	27.74	7.00	5.23%
Bemba	bem	Bantu ZoneM	Reading	116.32	11.43	7.41%

Table 1: Dataset descriptive statistic

with over 1400 languages, it provides extensive linguistic coverage and adaptability for low-resource settings. In addition, its lightweight design, with fewer parameters than other checkpoints, ensures faster and more efficient performance.

Language Model: We utilize 5-gram LMs for word and subword tokenization, and 10-gram LMs for character tokenization. For transformer-based LMs, we employ GPT-2 tailored to causal language modelling tasks[5]. The vocabulary sizes vary based on the tokenization approach: the number of distinct words for word-level, 2000 tokens for subword-level, and the number of distinct characters for character-level tokenization. These configurations are based on insights from preliminary experiments.

Pre- & Post-processing: We investigate LM integration across various tokenization levels and adapt ASR modelling accordingly. Training labels are generated by preprocessing transcripts into string sequences, embedding tokenization details directly into the training pipeline, as described in Section 2.2. This method allows the ASR model to produce outputs consistent with the chosen tokenization level. After prediction, post-processing is used to reverse the encoding steps and reconstruct the original sentence.

4 Results and Analysis

4.1 Fine-grained Tokenization Benefits

We experiment with different tokenization granularity with N-gram LMs. As shown in Table 2, compared with the coarse word-level tokenization, fine-grained tokenization improves performance for Kichwa, Mboshi, Japhug, and Bemba with Relative Word Error Rate (Relative WER) reduction of 6.5%, 7.3%, 8.4% and 9.8%, respectively. However, for Khinalug, the fine-grained approach shows comparable results but no clear gains, likely due to limited data and the spontaneous nature of the audio source.

Besides, we find the character level tokenization leads to the best performance for most languages, indicating character tokenization as a more effective choice. Regarding the outlier Mboshi, we notice its character ASR model struggles due to fast speaking speed or morphological complexity (Appendices A), complicating direct comparisons with subword models. Despite this challenge, the character-based approach shows greater relative improvements when transitioning from no LM to LM integration compared to the subword approach.

	No LM	Word	Subword	Char
Khinalug	42.2	**34.2**	37.9	35.8
Kichwa	17.7	15.4	15.3	**14.4**
Mboshi	31.4	27.3	**25.3**	30.1
Japhug	26.5	23.6	24.0	**21.3**
Bemba	40.0	38.6	35.5	**34.8**

Table 2: Experimental results for integrations granularity with N-gram LMs. Word, subword and char indicate the tokenization granularity. The evaluation metric is WER.

	No LM	N-gram	Transformer
Khinalug	45.5	35.9	40.5
Kichwa	18.6	**15.0**	17.1
Mboshi	33.4	**27.5**	28.5
Japhug	26.8	23.0	**21.8**
Bemba	39.0	**36.3**	37.2

Table 3: Experimental results for comparison between N-gram and transformer-based LMs. The resulted WER represents the average across experiments using word, subword, and character tokenization.

4.2 N-gram Integration Outperforms

Transformer-based LMs demonstrate notable strengths in perplexity evaluation, as detailed in

[5] https://huggingface.co/docs/transformers/tasks/language_modeling

	Text	WER	N-gram PPL	Trans PPL
Gold	alcaldesa juzgadomanta llukshikta rikukuni	-	7.5	5.4
No LM	alcaldesa husgadomanta llukshikta rikukuni	25.0	9.3	5.3
N-gram	alcaldesa juzgadomanta llukshikta rikukuni	0	7.5	5.4
Trans	alcaldesa huskadomanta llukshikta rikukuni	25.0	8.9	4.8

Table 4: An example of Kichwa with character-level tokenization is presented. Note that all hypotheses are considered during decoding in all experiments, but only one is selected as the final prediction with Equation 1 in each experiment.

Appendix B. We investigate transformer-based integration across all tokenization types and report the average scores. Surprisingly, as shown in Table 3, transformer-based LMs outperform N-gram LMs only for a single language, Japhug.

A closer examination of prediction samples reveals a misalignment between ASR performance and language modelling under the current integration approach. As shown in Table 4, the N-gram and transformer-based approaches do select the candidates with the lowest perplexity, and the perplexity values from transformer LM are indeed higher than that of N-gram LM, indicating the superior performance in causal language modelling. However, inconsistencies arise in how different LMs rank these candidates.

Specifically, the ASR gold transcript aligns more with the N-gram ranking than the transformer-based LM in this example. Although both models share the same vocabulary, allowing direct perplexity comparisons, their rankings might differ due to variations in architecture and evaluation. This suggests the current integration approach lacks robustness for low-resource languages, as it does not consistently improve ASR performance across models.

4.3 Language Optimization Matters

In developing ASR systems, prior research has predominantly focused on ASR training optimization, with limited attention to integrating LMs. In this study, we observe that the optimal tokenization granularity for five languages spans all three tokenization types and that the integration parameters vary significantly across languages. To highlight the importance of language-specific optimization, we experiment with reasonable parameter adaptation from Kichwa to Mboshi, which has a similar amount of supervised data, and Japhug, which has the same optimal tokenization type. As shown in Table 5, direct parameter transfer results in performance degradations of 32.0% and 43.2%, respectively.

	Token	(Alpha, Beta)	WER
Kichwa	char	(0.9, 5.0)	14.4
Mboshi	subword	(0.6, 2.0)	25.3
Transferred	char	(0.9, 5.0)	33.4
Japhug	char	(0.6, 1.0)	21.3
Transferred	char	(0.9, 5.0)	30.5

Table 5: Experiment results of parameter transferring from Kichwa to Mboshi and Japhug. Transferred means inferencing with the parameters optimized for Kichwa; Token indicates the tokenization type; Alpha and Beta indicate the parameters in decoding (Equation 1).

Moreover, we find that customizing beam size could improve inference speed while maintaining performance, demonstrating the practical benefits of tailored ASR systems (Appendix C.1). Additionally, our results indicate that ASR performance in low-resource languages is highly sensitive to training hyperparameters; even small adjustments in the learning rate can lead to significant performance differences (Appendix C.2). These findings emphasize the critical importance of language-specific settings in building effective ASR systems for low-resource languages.

5 Conclusion

This study focuses on improving contextual learning in ASR models for low-resource languages by examining tokenization granularity and the integration of transformer-based LMs. The findings show that fine-grained tokenization enhances ASR performance by addressing unknown words and increasing data usage efficiency. Moreover, integrating transformer-based LMs does not consistently outperform N-gram LMs in boosting ASR accuracy. Finally, our results indicate that directly applying experimental settings to new languages harms performance, emphasizing the importance of language-specific optimizations.

References

Panji Arisaputra, Alif Tri Handoyo, and Amalia Zahra. 2024. Xls-r deep learning model for multilingual asr on low-resource languages: Indonesian, javanese, and sundanese. *arXiv preprint arXiv:2401.06832*.

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

Pierre Godard, Gilles Adda, Martine Adda-Decker, Juan Benjumea, Laurent Besacier, Jamison Cooper-Leavitt, Guy-Noel Kouarata, Lori Lamel, Hélène Maynard, Markus Mueller, Annie Rialland, Sebastian Stueker, François Yvon, and Marcely Zanon-Boito. 2018. A very low resource language speech corpus for computational language documentation experiments. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

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, Cécile Macaire, Guillaume Jacques, Alexis Michaud, Benjamin Galliot, Maximin Coavoux, Solange Rossato, Minh-Châu Nguyên, and Maxime Fily. 2022. Fine-tuning pre-trained models for automatic speech recognition, experiments on a fieldwork corpus of japhug (trans-himalayan family). In *Proceedings of the Fifth Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 170–178, Dublin, Ireland. Association for Computational Linguistics.

Taiqi He, Kwanghee Choi, Lindia Tjuatja, Nathaniel Robinson, Jiatong Shi, Shinji Watanabe, Graham Neubig, David Mortensen, and Lori Levin. 2024. Wav2Gloss: Generating interlinear glossed text from speech. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 568–582, Bangkok, Thailand. Association for Computational Linguistics.

Yosuke Higuchi, Brian Yan, Siddhant Arora, Tetsuji Ogawa, Tetsunori Kobayashi, and Shinji Watanabe. 2022. BERT meets CTC: New formulation of end-to-end speech recognition with pre-trained masked language model. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5486–5503, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Eric Le Ferrand, Zoey Liu, Antti Arppe, and Emily Prud'hommeaux. 2024. Are modern neural ASR architectures robust for polysynthetic languages? In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 2953–2963, Miami, Florida, USA. Association for Computational Linguistics.

Zhaolin Li, Monika Rind-Pawlowski, and Jan Niehues. 2024. Speech recognition corpus of the khinalug language for documenting endangered languages. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 15171–15180, Torino, Italia. ELRA and ICCL.

Zoey Liu, Nitin Venkateswaran, Eric Le Ferrand, and Emily Prud'hommeaux. 2024. How important is a language model for low-resource ASR? In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 206–213, Bangkok, Thailand. Association for Computational Linguistics.

Ke-Han Lu and Kuan-Yu Chen. 2023. A context-aware knowledge transferring strategy for ctc-based asr. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 60–67.

Julia Mainzinger and Gina-Anne Levow. 2024. Fine-tuning ASR models for very low-resource languages: A study on mvskoke. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pages 76–82, Bangkok, Thailand. Association for Computational Linguistics.

Peter Mihajlik, Mate Kadar, Gergely Dosinszky, Yan Meng, Meng Kedalai, Julian Linke, Tibor Fegyo, and Katalin Mady. 2023. What kind of multi- or cross-lingual pre-training is the most effective for a spontaneous, less-resourced asr task? 2nd Annual Meeting of the Special Interest Group on Under-resourced Languages: SIGUL 2023 ; Conference date: 18-08-2023 Through 20-08-2023.

Vineel Pratap, Andros Tjandra, Bowen Shi, Paden Tomasello, Arun Babu, Sayani Kundu, Ali Elkahky, Zhaoheng Ni, Apoorv Vyas, Maryam Fazel-Zarandi, et al. 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.

Nay San, Martijn Bartelds, Blaine Billings, Ella de Falco, Hendi Feriza, Johan Safri, Wawan Sahrozi, Ben Foley, Bradley McDonnell, and Dan Jurafsky. 2023. Leveraging supplementary text data to kick-start automatic speech recognition system development with limited transcriptions. In *Proceedings of the Sixth Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 1–6, Remote. Association for Computational Linguistics.

Claytone Sikasote, Eunice Mukonde, Md Mahfuz Ibn Alam, and Antonios Anastasopoulos. 2023. BIG-C: a multimodal multi-purpose dataset for Bemba. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2062–2078, Toronto, Canada. Association for Computational Linguistics.

Chihiro Taguchi and David Chiang. 2024. Language complexity and speech recognition accuracy: Orthographic complexity hurts, phonological complexity doesn't. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15493–15503, Bangkok, Thailand. Association for Computational Linguistics.

Chihiro Taguchi, Jefferson Saransig, Dayana Velásquez, and David Chiang. 2024. Killkan: The automatic speech recognition dataset for kichwa with morphosyntactic information. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 9753–9763, Torino, Italia. ELRA and ICCL.

Silero Team. 2024. Silero vad: pre-trained enterprise-grade voice activity detector (vad), number detector and language classifier. https://github.com/snakers4/silero-vad.

A ASR Performance analysis

A.1 ASR Performance without Language Models

This section evaluates the performance of the ASR model using different tokenization methods without including language models (LMs). As outlined in Table 6, subword- and character-level tokenizations demonstrate slightly lower performance than word-level tokenization. This decline can be attributed to the added task of predicting the word boundary symbol "_." Nonetheless, this trade-off enables the incorporation of a more robust LM at the subword and character levels, enhancing the overall ASR performance during LM integration.

Lang	Word	Subwrod	Char
Khinalug	42.2	47.0	47.4
Kichwa	17.7	18.1	19.9
Mboshi	31.4	29.5	39.4
Japhug	26.5	26.5	27.5
Bemba	40.0	38.7	38.5

Table 6: ASR model performance of different tokenization types without LMs

A.2 Character Density Analysis

The Mboshi ASR model with character-level tokenization performs noticeably worse compared to word- and subword-level models. To investigate the outliers, we examine the character density of the corpus and find that the Mboshi corpus has a significantly higher number of characters per second than others, even though all audio files are sampled at 16 kHz (see Table 7).

We specifically use Voice Activity Detection (VAD) (Team, 2024) to measure the speaking duration and count the number of characters in the corresponding transcripts. We argue that the high character density negatively impacts character-level tokenization, as it leaves limited space for detecting separators between characters, resulting in information loss. Additionally, we suspect that the morphological complexity of Mboshi could be another contributing factor, but we are unable to evaluate this hypothesis due to a lack of linguistic expertise.

Lang	Train	Valid	Test
Khinalug	0.75	0.75	0.74
Kichwa	0.84	0.85	0.83
Mboshi	1.08	1.1	1.06
Japhug	0.83	0.84	0.84
Bemba	0.75	0.75	0.75

Table 7: Analytical statistic on character per second

B Causal Language Modelling

In this section, we compare N-gram and transformer-based language models (LMs) in the context of causal language modelling, which focuses on predicting the next token. This analysis supports our discussion in Section 4.2. As shown in Table 8, transformer-based LMs consistently achieve lower perplexity than N-gram LMs across all languages. This aligns with our expectation that transformer-based models outperform N-gram models in causal language modelling tasks due to their superior ability to capture contextual information. Additionally, we observe that larger datasets amplify the performance gap between the two types of models.

C Language Specific optimization

C.1 Integration Parameters

This section highlights the importance of language-specific parameters in language model integration.

	Word		Subword		Char	
	N-gram	Trans	N-gram	Trans	N-gram	Trans
Khinalug	1619.9	1243.2	709.5	604.7	10.3	8.8
Kichwa	1770.2	1271.7	550.2	313.7	6.9	4.0
Mboshi	1015.7	673.7	343.4	173.8	9.83	5.5
Japhug	699.6	448.0	181.7	75.1	7.3	3.5
Bemba	2915.9	1439.5	238.6	79.0	6.2	2.9

Table 8: Perplexity comparison for difference tokenization of N-gram and transformer-based LMs

	Beam	(α, β)	WER
Kichwa			
word	10	0.2/0	15.5
word	100	0.2/0	15.4
subword	10	0.9/5.0	15.7
subword	100	0.9/5.0	15.3
char	10	0.8/2.0	14.8
char	100	0.8/2.0	14.4
Japhug			
word	10	0/0	25.3
word	100	0.1/0	23.6
subword	10	0.1/2	24.2
subword	100	0.1/2	24.0
char	10	0.5/1	21.9
char	100	0.6/1	21.3

Table 9: Experimental results about beam searching and the selection of alpha and beta for Kichwa and Japhug

As illustrated in Table 9, a beam size of 10 performs comparably to a beam size of 100, demonstrating that this smaller value can reduce computational costs and hardware requirements. Additionally, we observe that the parameters alpha and beta require tailored values for optimal performance.

C.2 ASR Training Parameters

In this study, we explore various training hyperparameters to highlight their significance in low-resource scenarios. Specifically, we experiment with learning rates of 5e-4, 1e-4, 5e-5, 1e-5, 5e-6, and 1e-6. Our findings reveal that using the same hyperparameters across different languages or applying parameters optimized for one language to another results in noticeable performance degradation (as shown in Table 10). This underscores the importance of language-specific optimization when developing ASR systems for low-resource languages, in contrast to high-resource scenarios where the abundance of supervised data mitigates the influence of training hyperparameters.

Lang	Learning rate	CER	WER
Khinalug	1e-4	13.35	55.85
	1e-5	11.40	47.00
Japhug	1e-4	14.41	28.41
	1e-5	12.95	26.47

Table 10: Impact of learning rate on building ASR models