Boosting wav2vec2 Performance in Low-Resource Speech Recognition via Enhanced Pretraining and Fine-Tuning

Fonseka W.A.R.T
Department of Computer Science and Engineering
Faculty of Engineering, University of Moratuwa
Colombo, Sri Lanka
Email: thathsarana.21@cse.mrt.ac.lk

Uthayasankar Thayasivam
Department of Computer Science and Engineering
Faculty of Engineering, University of Moratuwa
Colombo, Sri Lanka
Email: ruthaya@cse.mrt.ac.lk

***Abstract* — Pretrained speech representation models such as Wav2Vec2 have achieved remarkable success in speech recognition. However, many downstream tasks such as low-resource language recognition, child voice recognition, and animal sound detection differ significantly from the pretraining objectives. In such cases, relying solely on fine-tuning pretrained models is often insufficient, while training from scratch is computationally expensive and data intensive. These challenges make it particularly difficult for researchers working with low-resource settings.**

In this work, we explore approaches to improve the efficiency of Wav2Vec2 for both pretraining and fine-tuning. We experimented with an inter-codebook similarity loss to enhance pretraining efficiency and investigated the use of residual quantization during fine-tuning to improve model adaptability with limited data. Our results demonstrate that these methods lead to faster convergence and improved performance in low-resource and domain-shifted scenarios, highlighting their potential for broader applications in speech and audio representation learning. The implementation of this work is publicly available at: https://github.com/RasaraThathsarana/In21-S7-CS4681-AML-Research-Projects.git

***Index Terms*—Wav2Vec2, Low-Resource Speech Recognition, Residual Vector Quantization, Inter-Codebook Similarity Loss**

I. Introduction

Speech recognition has seen remarkable advances in recent years, driven largely by self-supervised learning techniques that leverage large amounts of unlabeled audio data. Among these methods, Wav2Vec2 [1] has emerged as a state-of-the-art framework, learning rich speech representations during pretraining and enabling effective fine-tuning on various downstream tasks. These pretrained models have achieved impressive results in standard speech recognition, particularly for high-resource languages.

However, many practical applications like low-resource language recognition, child voice recognition, and animal sound detection are differ significantly from the data distributions used during pretraining. In such scenarios, simply fine-tuning pretrained models often fails to capture domain-specific acoustic or phonetic variations [2]. On the other hand, training models from scratch can yield better adaptability but requires large-scale datasets, extensive computation, and long training times, making it infeasible for researchers in resource-constrained environments [3].

Previous studies have explored techniques such as adapter modules, feature regularization, and domain-specific fine-tuning to mitigate these challenges, but issues related to data scarcity and computational inefficiency persist. Addressing these limitations is needed to democratize speech technology and enable its application across diverse and specialized domains.

In this work, we explore approaches to improve the efficiency and adaptability of Wav2Vec2 for both pretraining and fine-tuning. Specifically, we introduce an inter-codebook similarity loss(ICSL) to enhance pretraining efficiency and investigate the use of residual quantization(RVQ) to improve model adaptability with limited data. Our study provides practical strategies for efficient training of large speech models, aiming to reduce computational costs, specially focusing on reduce

training time while maintaining competitive performance in low-resource speech recognition tasks.

II. Background

A. Self-Supervised Speech Representation Learning

Self-supervised learning (SSL) has become the cornerstone of modern speech representation learning, enabling models to utilize vast amounts of unlabeled audio data for pretraining. Among SSL frameworks, Wav2Vec2.0 [1] has emerged as one of the most influential models, extending earlier works, Wav2Vec [4] and Contrastive Predictive Coding (CPC) [5]. It consists of three main components: a feature encoder that transforms raw audio into latent speech representations, a Transformer-based context network that captures long-range dependencies, and a quantization module that discretizes continuous latent features into codebook entries. During pretraining, certain timesteps are masked, and the model is trained to correctly identify quantized representations among distractors using a contrastive objective. This approach allows Wav2Vec2.0 to learn robust, context-aware speech representations that generalize across diverse acoustic conditions. See Fig. 1 for an overview of the Wav2Vec2.0 architecture.

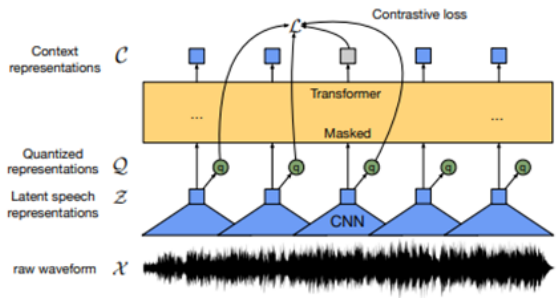

Fig. 1: Illustration of the Wav2Vec 2.0 framework, which jointly learns contextualized speech representations and an inventory of discretized speech units. (Adapted from the Wav2Vec 2.0 paper [1].)

Building on its success, several SSL variants have been developed to enhance performance and generalization. HuBERT [6] introduced an iterative pseudo-labeling process, where clustering-based targets guide pretraining and are refined over successive iterations to improve phonetic and linguistic structure capture. Data2Vec [7] generalized this paradigm to multiple modalities such as speech, text, and vision under a unified framework, while WavLM [8] focused specifically on speech, incorporating masked prediction and denoising objectives to strengthen acoustic and speaker representation learning. These models collectively demonstrate the rapid evolution of SSL techniques for speech representation but also highlight persistent challenges when adapting to low-resource or domain-shifted environments.

B. Low-Resource and Domain Adaptation Challenges

Despite the progress in SSL, many real-world speech tasks such as low-resource language recognition, child voice recognition [9], and animal vocalization detection are differ significantly from the data and domains used during pretraining. Fine-tuning large pretrained models directly on such datasets often fails to capture task-specific acoustic nuances, leading to degraded performance [10]. Techniques such as transfer learning and data augmentation have been employed to mitigate these issues, but their effectiveness diminishes as the domain gap widens. Cross-lingual pretraining approaches, such as XLSR [11], have shown promise by jointly training on multiple languages, though they remain computationally intensive and still struggle in extreme low-resource conditions.

C. Efficient Model Training Techniques

To reduce computational and data requirements, several strategies have been proposed to make SSL models more efficient. Knowledge distillation and model pruning have been used to compress large architectures while retaining representational quality [12]. Lightweight models such as DistilHuBERT [13] and LightHuBERT [14] demonstrate that smaller variants of SSL models can achieve comparable performance through distillation and architectural optimization. Meanwhile, adapter-based fine-tuning methods, such as Speech Adapter [15] and BitFit [16], allow selective parameter updates during adaptation, significantly reducing training cost and memory consumption. Additionally, quantization techniques have been explored to minimize model size and inference time, particularly beneficial for resource-constrained environments [17].

D. Residual Vector Quantization (RVQ)

Residual Vector Quantization (RVQ) [18] extends traditional vector quantization by using multiple codebooks in a hierarchical manner to achieve finer representation granularity. In standard quantization, an input vector is approximated using a single codebook entry, which may lead to high reconstruction error if the codebook is small.

RVQ addresses this limitation by sequentially applying several codebooks: the first provides a coarse quantization of the input, and each subsequent codebook encodes the residual error which is the difference between the input and the reconstructed vector from the previous stage. This multi-stage process iteratively refines the representation, achieving higher fidelity and compactness with fewer total codebook entries.

RVQ has been successfully applied in areas such as image compression and audio generation, where high-dimensional data must be efficiently represented. Its hierarchical encoding structure allows better preservation of subtle signal variations. When integrated into speech SSL models like Wav2Vec2, RVQ can enhance representation adaptability and robustness during fine-tuning, particularly in low-resource settings where data diversity is limited.

E. Summary and Motivation

The advances discussed above have greatly improved speech SSL models in terms of performance, generalization, and efficiency. However, efficiently adapting large pretrained models like Wav2Vec2 to low-resource or domain-specific tasks remains challenging. Both pretraining and fine-tuning demand significant computational resources, limiting accessibility for researchers with constrained hardware.

Our work builds upon these foundations by proposing two complementary strategies: an inter-codebook similarity loss(ICSL) to enhance pretraining efficiency, and the integration of residual vector quantization(RVQ) during fine-tuning to improve adaptability with limited data. These approaches aim to reduce convergence time in training process while maintaining competitive performance, contributing toward more accessible and efficient speech representation learning.

III. Methodology

This section outlines the dataset used in our study, experimental setup and the two key approaches we employed to enhance the efficiency of Wav2Vec 2.0: improvements during the pretraining stage and the fine-tuning stage. We also describe the model configurations adopted in our experiments, along with the evaluation metrics used to assess performance. In addition, we highlight specific design choices made to adapt the framework for low-resource scenarios, ensuring a fair and meaningful evaluation.

A. Dataset

For our experiments, we used the LibriSpeech dataset [19], which was also used in the original Wav2Vec 2.0 research. To mimic a low-resource setting, we selected a 10-hour subset from the full 960-hour corpus. We specifically used the cleaned subset to ensure higher data quality.

When constructing the 10-hour dataset, we only included utterances longer than 5 seconds, since very short speech segments often caused issues during the masking step in pretraining. After filtering, our training set contained 2,850 utterances.

For evaluation, we used the original LibriSpeech validation set. Applying the same filtering rule (removing utterances shorter than 5 seconds), the final validation set consisted of 2,703 utterances.

B. Proposed Approaches

To improve the efficiency of Wav2Vec 2.0 in low-resource scenarios, we explored two key strategies focusing on both pretraining and fine-tuning stages.

1. Pretraining Enhancement:

In self-supervised learning models that rely on codebook-based vector quantization. There is a potential issue where all vectors within a single codebook may collapse into a single representation. The original Wav2Vec 2.0 addresses this by introducing a diversity loss term. It encourages uniform usage of codebook entries and helps prevent intra-codebook collapse. The diversity loss is defined as

$$\mathcal{L}_{\text{diversity}} = \frac{1}{G} \sum_{g=1}^{G} \left(1 - \frac{H(\bar{p}_g)}{\log V} \right) \tag{1}$$

where G is the number of codebooks, V is the number of entries in each codebook, and $H(\bar{p}_g) = -\sum_{i=1}^{V} \bar{p}_{g,i} \log \bar{p}_{g,i}$ represents the entropy of the average soft assignment probabilities for codebook g. This term maximizes entropy to ensure uniform usage of codebook entries and prevents intra-codebook collapse.

However, when multiple codebooks are used, another issue can arise called inter-codebook similarity, which is distinct codebooks learn overlapping or redundant representations. This reduces the diversity and effectiveness of quantized vectors, limiting the model's representational capacity. The original diversity loss does not explicitly handle this kind of redundancy. So this may lead to inefficient feature learning.

To mitigate this issue, we propose an additional loss term called the Inter-Codebook Similarity Loss (ICSL).

This loss measures the cosine similarity between embeddings across multiple codebooks and penalizes high similarity. This encourages each codebook to capture distinct aspects of the input signal. It is formulated as

$$\mathcal{L}_{\text{ICSL}} = \frac{1}{G(G-1)} \sum_{i=1}^{G} \sum_{j=i+1}^{G} \text{mean}\left(\cos_{\text{sim}}(E_i, E_j)\right) \tag{2}$$

where Ei and Ej denote the embedding matrices of codebooks i and j, and $\cos\_sim(E_i, E_j) = \frac{E_i \cdot E_j^T}{\|E_i\| \|E_j\|}$ represents the cosine similarity between their embeddings. Penalizing this similarity encourages greater diversity between codebooks.

The total pretraining objective thus combines the standard contrastive loss, diversity loss, and the proposed ICSL term as follows:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{contrastive}} + \lambda_d \mathcal{L}_{\text{diversity}} + \lambda_s \mathcal{L}_{\text{LCSL}} \tag{3}$$

where λ_d and λ_s are weighting coefficients controlling the contribution of each regularization term. The contrastive loss, which forms the main self-supervised objective of *Wav2Vec 2.0*, is expressed as

$$\mathcal{L}_{\text{contrastive}} = -\sum_{t \in M} \log \frac{\exp\left(\frac{\text{sim}(c_t, q_t)}{\kappa}\right)}{\sum_{\tilde{q} \in Q_t} \exp\left(\frac{\text{sim}(c_t, \tilde{q})}{\kappa}\right)} \tag{4}$$

where M denotes the masked time steps, c_t is the contextualized representation, q_t is the true quantized vector, Q_t is the set of one positive and K negative quantized samples, and κ is the temperature parameter.

By promoting both intra- and inter-codebook diversity, the total loss encourages richer and more discriminative representations, stabilizing pretraining and improving the model's feature learning efficiency.

Previous research on Wav2Vec 2.0 noted that using a large number of codebooks can degrade model performance. This may happen possibly due to such inter-codebook collapse. Through this work, we empirically explore this phenomenon and investigate whether incorporating ICSL can counteract this effect and improve training efficiency.

This approach allows us to examine the impact of inter-codebook regularization under different codebook configurations. The empirical results and comparative analysis of these experiments are presented in Section IV.

2. Fine-Tuning Enhancement

In the original *Wav2Vec 2.0* framework, fine-tuning is typically performed by attaching a task-specific head, such as a linear projection layer, on top of the pretrained model and optimizing it using a supervised loss such as the Connectionist Temporal Classification (CTC) loss. While this approach performs well for large and diverse datasets, it may be less effective in low-resource scenarios, where data scarcity can limit the model's ability to adapt effectively to the downstream task.

To address this limitation, we introduce a Residual Vector Quantization (RVQ) representation between the *Wav2Vec 2.0* embedding layer and the task-specific head during fine-tuning. The RVQ module refines the latent speech representations by applying a series of residual quantizers, where each subsequent quantizer encodes the residual error left by the previous one.

Given an input latent feature vector x, the first quantizer produces a coarse quantized approximation q_1, and the residual is computed as:

$$r_1 = x - q_1 \tag{5}$$

For each subsequent quantizer $l \in \{2, 3, \ldots, L\}$, the residuals are recursively encoded as:

$$r_l = r_{l-1} - q_l \tag{6}$$

where q_l is the quantized output of the l-th residual quantizer. The final refined latent representation is obtained by summing the quantized outputs from all quantization stages:

$$\hat{x} = \sum_{l=1}^{L} q_l \tag{7}$$

This hierarchical quantization process allows the model to capture fine-grained variations in the latent space and minimizes representational redundancy, enabling more efficient adaptation with limited data.

During fine-tuning, the model is optimized using the Connectionist Temporal Classification (CTC) loss, which aligns the model's output sequence with the target transcription without requiring explicit frame-level alignment. The CTC loss is expressed as:

$$\mathcal{L}_{\text{CTC}} = -\log \sum_{\pi \in B^{-1}(y)} P(\pi \mid x) \tag{8}$$

where π represents all possible alignments of the target label sequence y with the input x, and B is the collapsing function that merges repeated symbols and removes blank tokens.

By incorporating the RVQ module, the effective feature representation $\hat{x}$ replaces the original embedding

x in the fine-tuning process, leading to the following overall fine-tuning objective:

$$\mathcal{L}_{\text{fine-tune}} = \mathcal{L}_{\text{CTC}} \tag{9}$$

where the quantized representation $\hat{x}$ improves convergence speed and representation quality by encoding compact and discriminative speech features.

In this study, we used the Hugging Face pretrained *Wav2Vec 2.0 Base* model trained on the 960-hour LibriSpeech corpus as the base model. Fine-tuning was conducted using our filtered 10-hour LibriSpeech subset, employing the CTC loss for optimization. By integrating RVQ into this process, we observed improved convergence rates and enhanced adaptability of the model under data-constrained conditions.

C. Experimental Setup

All experiments were conducted using a single NVIDIA P100 GPU with 16 GB of memory provided by the Kaggle environment. The main configurations used for pretraining and fine-tuning are summarized in Table I.

TABLE I: Pretraining and Fine-Tuning Configurations

Parameter	Pretraining	Fine-Tuning
Model	Base (random init)	Base (pretrained)
Processor	facebook/wav2vec2-base	facebook/wav2vec2-base
Batch Size	16	8
Grad. Accum. Steps	4	8
Learning Rate	1×10^{-4}	1×10^{-4}
Scheduler	Linear	Linear
Total Steps	2200	1500
LR Warm-up Ratio	0.1	0.2
LR Decay Ratio	0.9	0.6

For pretraining, the Inter-Codebook Similarity Loss (ICSL) was added to the total loss function with a weight of 0.1 to encourage diversity across multiple codebooks.

D. Evaluation Metrics

Different metrics were employed for pretraining and fine-tuning to align with their objectives:

Pretraining: Evaluated using Contrastive Loss, which measures the model's ability to distinguish between correct and incorrect quantized targets. Lower values indicate better alignment between latent and discrete representations.

Fine-Tuning: Evaluated using Connectionist Temporal Classification (CTC) Loss and Word Error Rate (WER). CTC Loss measures prediction alignment quality, while WER reflects end-to-end transcription accuracy (lower is better).

IV. Experiments

In this section, we present the experimental evaluation of our proposed enhancements to Wav2Vec 2.0 under low-resource conditions. Both pretraining and fine-tuning experiments were conducted using the 10-hour LibriSpeech subset described in Section III.A.

A. Pretraining Experiments

We evaluated the effectiveness of the Inter-Codebook Similarity Loss (ICSL) in improving Wav2Vec 2.0 pretraining across different codebook configurations. Specifically, we compared models using 2 codebooks and 8 codebooks, both with and without the ICSL regularization term. The contrastive loss was recorded for both the training and validation sets. Figures 2 and 3 illustrate the corresponding training and validation loss curves.

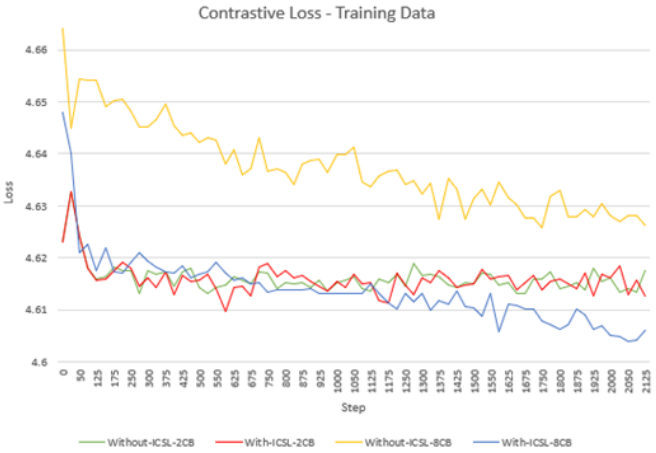

Fig. 2: Training contrastive loss curves for 2 and 8 codebook configurations, demonstrating faster and more stable convergence with ICSL for the 8-codebook setup, lower convergence without ICSL for 8 codebooks, and similar performance across both settings when using 2 codebooks.

However, the 8-codebook configuration without ICSL showed the slowest convergence, indicating that simply increasing the number of codebooks without addressing inter-codebook redundancy can degrade learning efficiency. These results suggest that ICSL effectively mitigates inter-codebook redundancy, allowing the model to benefit from richer quantization while maintaining

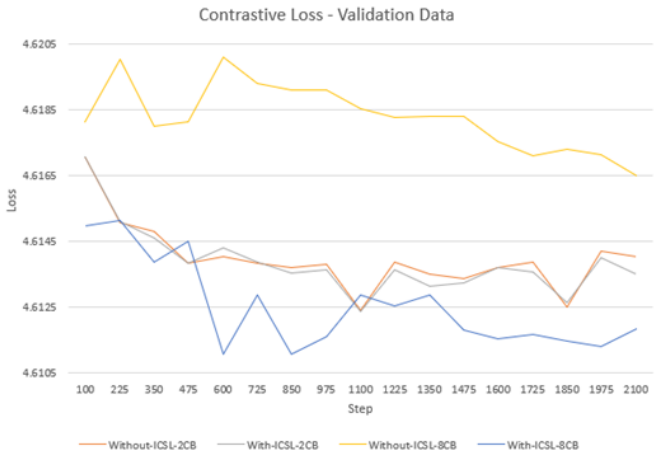

Fig. 3: Validation contrastive loss variation across different configuration.

training stability. Consequently, increasing the number of codebooks can enhance representation learning only when inter-codebook regularization is applied.

In prior work, increasing the number of codebooks did not lead to better performance, likely due to inter-codebook collapse. By incorporating the Inter-Codebook Similarity Loss (ICSL), we mitigate this issue, enabling multiple codebooks to contribute more effectively to representation learning. To validate the effectiveness of our ICSL regularization, Figure 4 presents a comparison of the contrastive loss after 2200 update steps for models trained with and without ICSL using 2, 4, 8, and 16 codebooks.

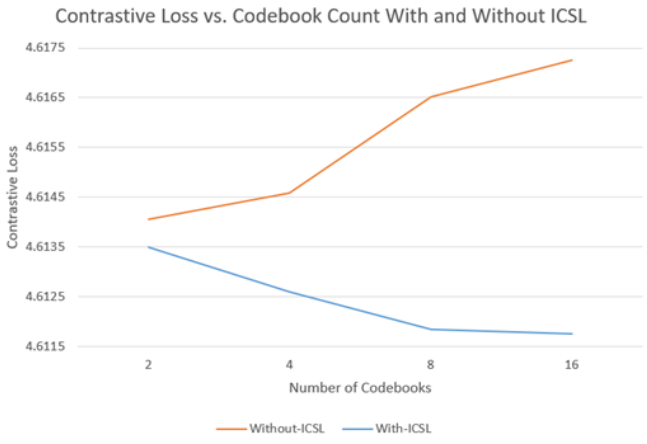

Fig. 4: Contrastive loss comparison across different codebook configurations with and without ICSL after 2200 update steps.

Figure 4 clearly shows that as the number of codebooks increases, the configuration with ICSL exhibits a significant reduction in contrastive loss, while the configuration without ICSL shows an increase in loss. This suggests that ICSL effectively prevents codebook redundancy, enhances representation learning efficiency, and addresses a key limitation of the original Wav2Vec 2.0 which is codebook collapse when using multiple codebooks.

Fine-Tuning Experiments

To evaluate the effectiveness of Residual Quantization Vectors (RVQ) in low-resource scenarios, we fine-tuned the pretrained Wav2Vec 2.0 base model on the same 10-hour LibriSpeech subset. Two configurations were tested: 1. Baseline: Pretrained Wav2Vec 2.0 with a standard classification head. 2. RVQ-enhanced: Pretrained Wav2Vec 2.0 with RVQ inserted between the embedding layer and the classification head.

Training was performed using CTC loss as the objective function, and both models were evaluated on training and validation subsets. Figures 5 and 6 illustrate the CTC loss curves for training and validation, respectively. For the initial model, we used four quantization levels for RVQ.

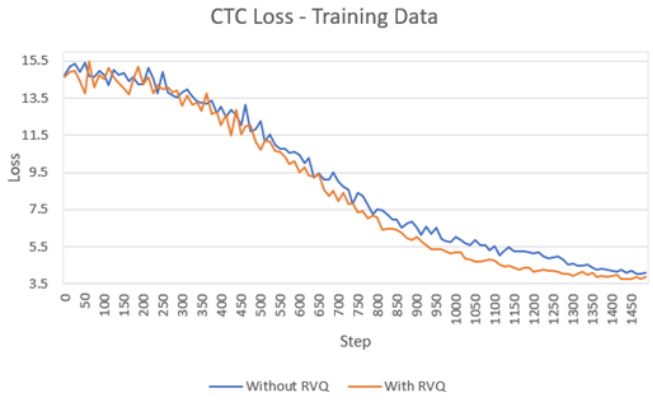

Fig. 5: Training CTC loss for models with and without RVQ, showing faster convergence with RVQ integration.

Results show that the RVQ-enhanced model converges faster than the baseline in both training and validation sets. This suggests that in low-resource conditions, residual quantized representations provide more distinct and effective features than directly using the continuous embeddings. Faster convergence is particularly beneficial for low-resource settings where computational resources are limited, as it reduces training time while maintaining performance potential. To evaluate the impact of Residual Vector Quantization (RVQ) depth on fine-tuning

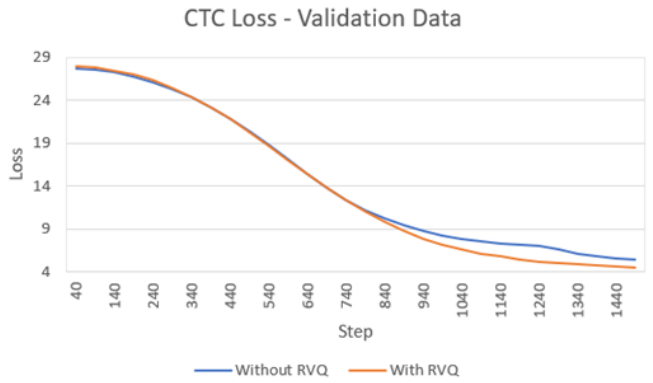

Fig. 6: Validation CTC loss for models with and without RVQ, indicating improved convergence with RVQ.

efficiency, we tested models with different numbers of quantization levels: 1, 2, 4, 8, and 16. Each model was fine-tuned for 1,500 steps using the same configuration described in Section III.C. Figure 7 presents the relationship between the number of RVQ quantization levels and the resulting CTC loss.

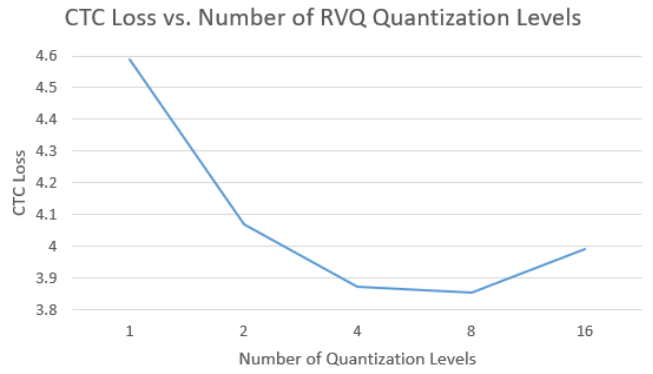

Fig. 7: Effect of RVQ quantization levels on CTC loss during fine-tuning.

Results show that increasing the number of quantization levels generally improves model performance up to a certain point. Specifically, the CTC loss decreases steadily from 4.59 at one level to 3.85 at eight levels, indicating enhanced representational refinement and better adaptation to limited data. However, at sixteen levels, the loss slightly increases to 3.99, suggesting potential over-quantization or redundancy at deeper quantization hierarchies.

These findings demonstrate that a moderate number of RVQ quantization levels (around four to eight) achieves the best balance between representational capacity and training stability in low-resource fine-tuning scenarios; however, this may change depending on data availability and the nature of the data. Using the appropriate RVQ quantization level can accelerate training convergence because the classification head learns from distinct code vectors rather than continuous values, making the training process faster and more efficient. This setup is particularly useful for researchers and developers working with limited computational resources.

V. Conclusion

This study presented methods to enhance the efficiency and robustness of Wav2Vec 2.0 in low-resource speech recognition. We introduced the Inter-Codebook Similarity Loss (ICSL) to encourage diversity across multiple codebooks during pretraining and incorporated Residual Vector Quantization (RVQ) during fine-tuning to improve convergence speed and adaptability. Experimental evaluations on the 10-hour LibriSpeech subset demonstrated that ICSL with multiple codebooks accelerates convergence and enhances representational diversity, while RVQ improves fine-tuning efficiency and model stability. Future research will explore the integration of RVQ into the pretraining phase and the application of these techniques to diverse low-resource languages and specialized speech domains. Overall, the proposed approaches provide a promising direction for building more data-efficient and computationally lightweight speech models.

References

[1] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *arXiv preprint arXiv:2006.11477*, 2020.

[2] T. Reitmaier, et al., "Opportunities and challenges of automatic speech recognition systems for low-resource language speakers," in *CHI Conference on Human Factors in Computing Systems*, 2022.

[3] L. Lugo and V. Vielzeuf, "Towards efficient self-supervised representation learning in speech processing," (preprint / technical report), 2021.

[4] S. Schneider, A. Baevski, R. Collobert, and M. Auli, "wav2vec: Unsupervised pre-training for speech recognition," *arXiv preprint arXiv:1904.05862*, 2019.

[5] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[6] W.-N. Hsu, et al., "HuBERT: Self-supervised speech representation learning by masked

prediction of hidden units," *arXiv preprint arXiv:2106.07447*, 2021.

[7] A. Baevski, W.-N. Hsu, Q. Xu, A. Babu, J. Gu, and M. Auli, "data2vec: A general framework for self-supervised learning in speech, vision and language," *arXiv preprint arXiv:2202.03555*, 2022.

[8] S. Chen, et al., "WavLM: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, 2022.

[9] A. Potamianos, S. Narayanan, and S. Lee, "Automatic speech recognition for children," in *Eurospeech*, 1997.

[10] C. Yu, M. Kang, Y. Chen, J. Wu, and X. Zhao, "Acoustic modeling based on deep learning for low-resource speech recognition: An overview," *IEEE Access*, 2020.

[11] A. Babu, et al., "XLS-R: Self-supervised cross-lingual speech representation learning at scale," *arXiv preprint arXiv:2111.09296*, 2021.

[12] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[13] H.-J. Chang, S. Yang, and H. Lee, "DistilHuBERT: Speech representation learning by layer-wise distillation of hidden-unit BERT," *arXiv preprint arXiv:2110.01900*, 2022.

[14] R. Wang, et al., "LightHuBERT: Lightweight and configurable speech representation learning with once-for-all Hidden-Unit BERT," *arXiv preprint arXiv:2203.15610*, 2022.

[15] B. Thomas, S. Kessler, and S. Karout, "Efficient adapter transfer of self-supervised speech models for automatic speech recognition," *arXiv preprint arXiv:2202.03218*, 2022.

[16] E. B. Zaken, S. Ravfogel, and Y. Goldberg, "BitFit: Simple parameter-efficient fine-tuning for transformer-based masked language-models," *arXiv preprint arXiv:2106.10199*, 2021.

[17] Y. Guo, "A survey on methods and theories of quantized neural networks," *arXiv preprint arXiv:1808.04752*, 2018.

[18] N. Zeghidour, A. Luebs, A. Omran, J. Skoglund, and M. Tagliasacchi, "SoundStream: An end-to-end neural audio codec," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2021.

[19] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *ICASSP*, 2015.