HASRD: Hierarchical Acoustic and Semantic Representation Disentanglement

Hussein, Amir; Khurana, Sameer; Wichern, Gordon; Germain, François G; Le Roux, Jonathan

TR2025-122   August 20, 2025

Abstract

Effective speech representations for spoken language models must balance semantic relevance with acoustic fidelity for high- quality reconstruction. However, existing approaches struggle to achieve both simultaneously. To address this, we introduce Hierarchical Acoustic and Semantic Representation Disentanglement (HASRD, pronounced "hazard"), a framework that factorizes self-supervised learning representations into discrete semantic and acoustic tokens. HASRD assigns the semantic representation to the first codebook, while encoding acoustic residuals in subsequent codebooks. This preserves ASR performance while achieving high-quality reconstruction. Addi- tionally, we enhance HASRD's encoder efficiency, improving ASR performance without compromising reconstruction quality. Compared to SpeechTokenizer, HASRD achieves a 44% relative WER improvement, superior reconstruction quality, and 2x lower bitrate, demonstrating its effectiveness in disentangling acoustic and semantic information.

Interspeech 2025

HASRD: Hierarchical Acoustic and Semantic Representation Disentanglement

Amir Hussein[1,2], Sameer Khurana[1], Gordon Wichern[1], Francois G. Germain[1], Jonathan Le Roux[1]

[1]Mitsubishi Electric Research Laboratories, USA
[2]Johns Hopkins University, USA

ahussei6@jhu.edu, {wichern,germain,leroux}@merl.com

Abstract

Effective speech representations for spoken language models must balance semantic relevance with acoustic fidelity for high-quality reconstruction. However, existing approaches struggle to achieve both simultaneously. To address this, we introduce Hierarchical Acoustic and Semantic Representation Disentanglement (HASRD, pronounced "hazard"), a framework that factorizes self-supervised learning representations into discrete semantic and acoustic tokens. HASRD assigns the semantic representation to the first codebook, while encoding acoustic residuals in subsequent codebooks. This preserves ASR performance while achieving high-quality reconstruction. Additionally, we enhance HASRD's encoder efficiency, improving ASR performance without compromising reconstruction quality. Compared to SpeechTokenizer, HASRD achieves a 44% relative WER improvement, superior reconstruction quality, and 2x lower bitrate, demonstrating its effectiveness in disentangling acoustic and semantic information.

Index Terms: self-supervised learning, speech recognition, vector quantization, neural audio codecs

1. Introduction

The remarkable performance of large language models has inspired the development of spoken language models (SLMs) that leverage discrete speech representations [1–3]. SLM use cases range from cross-modal conversational abilities, enabling them to understand and generate multimodal content, to various speech-related tasks such as automatic speech recognition (ASR), speech translation (ST), and spoken language understanding [3, 4]. Discrete speech representations for SLMs are broadly categorized into semantic and acoustic tokens [2, 5]. Prior work [5, 6] highlights the importance of balancing semantic content and acoustic fidelity to support both language modeling and high-quality synthesis. Semantic tokens are derived by pre-training a transformer encoder via self-supervised learning (SSL) typically based on masked prediction [7–9], followed by vector quantization. While effective for ST and ASR [6, 10, 11], semantic tokens often degrade synthesis quality and lack fine acoustic details compared to neural audio codecs [5, 12]. In contrast, acoustic tokens are produced by neural audio codecs trained for reconstruction [13, 14], which use hierarchical quantization such as residual vector quantization (RVQ) to preserve audio fidelity [15, 16], but lack semantic structure. This trade-off motivates the need for a unified architecture that supports both accurate recognition and high-quality resynthesis.

Recently, researchers proposed jointly learning semantic and acoustic discrete representations by distilling semantic in-

formation into the first codebook of a neural codec, while using the remaining codebooks for acoustic reconstruction [5, 15, 17]. While this improves semantic content, performance on semantic tasks still degrades compared to the teacher model. Moreover, state-of-the-art codecs [16] achieve high reconstruction quality by using codebooks with small latent dimensionalities, optimizing codebook utilization. However, this design makes distillation within such constrained latent spaces particularly challenging. A closely related work [18], developed concurrently with ours, focuses on voice conversion by disentangling content (K-means quantized SSL) from speaker variation (SSL residual after content subtraction). However, it assumes that a single SSL layer captures both content and acoustic information, and it remains unclear how well the learned representation performs on semantic tasks such as ASR.

To address the aforementioned limitations, we propose a unified model that disentangles and jointly learns semantic and acoustic speech tokens. Unlike prior work, our approach aims to preserve performance on semantic tasks while achieving high-quality reconstruction. To this end, we introduce a novel hierarchical disentanglement framework inspired by the concept of RVQ. Prior research has demonstrated that SSL representations are well-suited for semantic tasks and can also facilitate speech resynthesis [19–22]. Motivated by this, we factorize SSL representations into discrete semantic and acoustic tokens, where the semantic layer forms the first RVQ codebook, and the remaining codebooks encode the acoustic residual via a reconstruction objective. Our key contributions are: 1) HASRD, a novel framework for learning disentangled acoustic and semantic representations, 2) an efficient CNN encoder that reduces computational complexity while maintaining high reconstruction quality in neural codecs, and 3) an improved design of SSL that utilizes random projection quantization, leading to better quantized representations. Additionally, we provide an in-depth analysis of information disentanglement, highlighting the conflicting objectives of acoustic and semantic representations and explaining how optimizing one often degrades the other.

2. Proposed Approach

For discrete speech representations to be effective for SLMs, they must capture semantic information while preserving acoustic details to enable high-quality speech synthesis. However, these objectives often conflict, as enhancing one can degrade the other, making it challenging to learn a single representation that balances both, as illustrated in Section 4.3. To overcome this challenge, we propose a two-stage approach that combines 1) SSL pretraining with a masked language modeling (MLM) objective and 2) continued training with residual vector quantization (RVQ) and reconstruction objective. This framework
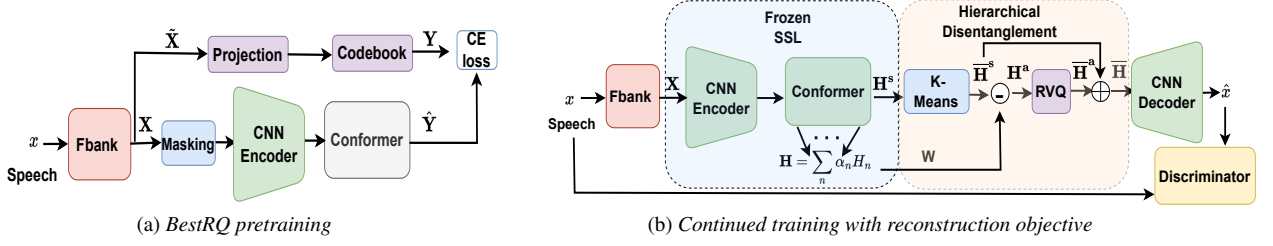

Figure 1: *Proposed HASRD framework for hierarchical disentangling of acoustic and semantic representations*

(a) *BestRQ pretraining* (b) *Continued training with reconstruction objective*

enables good word error rate (WER) performance, strong compression, and high-quality reconstruction.

2.1. Semantic Pre-Training

We adopt SSL pre-training with an MLM objective. Our CNN encoder design is inspired by the Descript Audio Codec framework [16]. Given computational constraints, we first enhance the efficiency of the CNN encoder. Instead of raw audio, our system first computes an F-dimensional log mel-spectrogram $\mathbf{X} \in \mathbb{R}^{T \times F}$ as input. The encoder consists of two 1D-CNN blocks with strided convolutions, each containing three residual units with 1D dilated convolutions. We use the Snake activation function [16] for nonlinearity. To further reduce the number of parameters, we employ depthwise separable convolutions [23], a special case of grouped convolution where the number of groups matches the number of channels. To better capture semantic information, we enhance the CNN encoder by stacking Conformer modules with multi-head attention (MHA) on top of the CNN blocks, and we pre-train it with an MLM objective. Following BestRQ [24], we apply a random projection quantizer to the stacked fbank feature sequence $\tilde{X} \in \mathbb{R}^{T' \times F'}$, where $T' = T/4$ and $F' = 4F$, with a single codebook to generate labels $\mathbf{Y}$ for masked speech features. The stacked features at each frame are quantized to the nearest codebook entry using the standard cosine-distance-based code lookup. Note that the time resolution T' matches the output of the CNN encoder in our SSL model. The SSL model is trained to predict labels for the masked regions using unmasked features as context, guided by quantizer-provided labels:

$$\mathcal{L}_{\text{MLM}} = - \sum_{m \in \mathcal{M}} \log P(y_m \mid \mathbf{X}_{\backslash \mathcal{M}}), \qquad (1)$$

where $\mathcal{M}$ denotes the set of masked positions and $\mathbf{X}_{\backslash \mathcal{M}}$ the unmasked features.

2.2. Hierarchical Information Disentanglement

It has been shown that the representations of specific SSL layers are well-suited for semantic tasks, while others are more effective for acoustic-related tasks such as speech resynthesis [11, 19, 20]. Motivated by this observation, we hypothesize that SSL representations can be factorized into semantic and acoustic representations. Let $\{\mathbf{H}_1, \ldots, \mathbf{H}_N\}$, where $\mathbf{H}_n \in \mathbb{R}^{U \times D}$, denote the latent representations with temporal dimension U (after CNN downsampling) and latent dimension D, obtained from the pre-trained SSL layers described in Section 2.1. Since it is not known which representation is best suited for learning acoustic tokens, we propose to introduce learnable weights that adaptively combine the representations as shown in Fig. 1. Specifically, we compute the full representation $\mathbf{H}$ as a convex combination $\mathbf{H} = \sum_{n=1}^{N} \alpha_n \mathbf{H}_n$, where α_n is a learned normalized scalar weight. We identify the index $n_s \in \{1, \ldots, N\}$ of the best performing representation for the ASR task, and de-

fine the corresponding representation $\mathbf{H}^s = \mathbf{H}_{n_s}$ as the semantic representation. The corresponding quantized semantic representation is then obtained as $\overline{\mathbf{H}}^s = \text{kmeans}(\mathbf{H}^s)$, where k-means is trained offline as described in Section 3. The acoustic representation $\mathbf{H}^a \in \mathbb{R}^{U \times D}$ is obtained by removing the semantic component from a projected version of the full representation $\mathbf{H}$ using a learnable transformation matrix $\mathbf{W} \in \mathbb{R}^{D \times D}$, intended to align the semantic part of $\mathbf{H}$ with $\overline{\mathbf{H}}^s$ for optimal removal:

$$\mathbf{H}^a = \mathbf{H}\mathbf{W} - \overline{\mathbf{H}}^s \qquad (2)$$

During the continued training stage, the encoder (CNN + Conformer) is frozen. To ensure that the decoder reconstruction relies on the semantic quantized representation $\overline{\mathbf{H}}^s$, we adopt a multistage hierarchical quantization approach inspired by RVQ. Specifically, we treat $\overline{\mathbf{H}}^s$ as the output of the first quantization codebook of an RVQ, and quantize the residual $\mathbf{H}^a$ obtained by Eq. (2) using subsequent RVQ codebooks, denoting by $\overline{\mathbf{H}}^a_m$ the output of each quantization stage m, $m = 1, \ldots, M - 1$. The final quantized representation is then computed as:

$$\overline{\mathbf{H}} = \overline{\mathbf{H}}^s + \sum_{m=1}^{M-1} \overline{\mathbf{H}}^a_m. \qquad (3)$$

The resulting $\overline{\mathbf{H}}$ is passed to the decoder, which mirrors the encoder's design but employs 1D transposed convolutions to progressively upsample the feature sequence into a waveform.

2.3. Discriminators and Reconstruction Loss Functions

Our approach adopts the discriminators and loss functions proposed in [16]. Specifically, we employ a combination of multiscale (MSD) [25] and multiperiod (MPD) waveform discriminators [26], alongside a complex short-time Fourier transform (STFT) discriminator operating at multiple timescales. To further enhance perceptual quality, we integrate a multiband, multiscale STFT discriminator with an L1 feature matching loss [16]. For reconstruction, we optimize an L1 loss on mel spectrograms, leveraging multiple window lengths and mel-bin configurations to capture both fine and coarse spectral details. Codebook learning combines commitment and codebook losses, with gradient propagation handled via the straight-through estimator.

3. Experimental Setup

Data & Pre-processing: We train our models on the LibriSpeech dataset [27], following [5]. For acoustic training, we adopt the DAC framework [16], extracting random 5-second segments (vs. 0.38 s in DAC) from each utterance, zero-padding shorter ones for uniform batch sizes. This extended duration captures richer context and improves semantic representation. Unless stated otherwise, we use nine codebooks: DAC employs nine RVQ codebooks, while HASRD combines one semantic k-means codebook with eight acoustic RVQ codebooks. The k-means model is trained offline on SSL latent representations

Table 1: *Impact of the improved CNN encoder on reconstruction quality (mel distance, STFT distance, and perceptual quality via VISQOL) and computational complexity (reduction of GPU memory usage and improvement of inference speed)*

Model	Segment	Bitrate	Train	Params	GPU-Mem	Inference	Mel ↓	STFT ↓	ViSQOL ↑
DAC	0.4 s	2.79 kbps	100 h	76.6M	-	-	0.75	1.55	4.22
DAC (baseline)	5.0 s	2.79 kbps	100 h	76.6M	-	-	**0.66**	**1.45**	4.45
DAC+fbank	5.0 s	2.79 kbps	100 h	68.0M	−25%	x2	**0.66**	1.50	**4.50**
DAC+fbank+SepConv	5.0 s	2.79 kbps	100 h	48.0M	−25%	x3	0.68	1.52	4.45

using 10% of the train-clean-100 subset. For the ablation study in Section 4.1, models are trained on the 100h train-clean-100 subset for 30K iterations; all other experiments use the full 960h LibriSpeech dataset with 100K iterations. For SSL training, data is prepared using Lhotse [28], enabling dynamic batching by total batch duration. Audio is converted to 80-dimensional mel-spectrograms via TorchAudio's Kaldi filterbank[1], using a 25 ms window with 8 ms frame shift.

Encoder: Our encoder consists of a CNN module described in Section 2.1 followed by Conformer layers for enhanced semantic modeling. The CNN module consists of two 1D-CNN blocks, each with a kernel size of 4, a stride of 2, and a latent dimension of 1024. This results in an overall downsampling factor of $4 \times 160 = 512$, where the factor of 160 originates from the 8 ms frame shift during feature extraction. This matches the downsampling of the DAC encoder. Each CNN block contains three residual units, each with a 1D-CNN and dilation rates of $\{1, 3, 9\}$. The Conformer module follows [29], with modifications to reduce parameters due to GPU memory constraints. Specifically, we use 12 layers instead of 24 as in [24], increase the number of attention heads from 8 to 12, and reduce both the feedforward dimension from 2048 to 1024 and the attention dimension from 768 to 512. The encoder is pre-trained on the full LibriSpeech dataset using the MLM objective. In our proposed approach, we found that layer $n_s = 8$ yields the best ASR performance, which we use as the semantic representation $\mathbf{H}^s$.

Decoder: The decoder follows [16], consisting of 4 1D-CNN upsampling layers with strides [8, 8, 4, 2] and a latent dimension of 1536. We also adopt the same multi-period and multi-scale STFT discriminators from [16]. Acoustic training uses a batch size of 8, while MLM training uses a batch duration of 600 seconds per GPU across 8 A40 GPUs. SSL training is optimized with AdamW, a learning rate of 0.0008, a Noam scheduler with 25K warmup steps, and gradient accumulation of 2. Reconstruction training is optimized with AdamW, a learning rate of 1×10^{-4} and exponential learning rate schedule.

Evaluation: We use ASR as the downstream task to evaluate the quality of the semantic tokens learned by our model. We use a 2-layer BLSTM with a hidden dimensionality of 1024 as our ASR encoder. It is trained using tuples (x, y), where x is the learned continuous or discrete representation outputted by the pre-trained models, and y is the character transcript. The model is trained using CTC [30] on the train-clean-100 subset of LibriSpeech. We train the model for 30K iterations using the Adam optimizer with a learning rate of 1e-4 and a batch size of 32. We report the word error rate (WER) and character error rate (CER) on the LibriSpeech test-clean subset.

For reconstruction evaluation, we compute the loss using log-mel spectrogram distances across window lengths [32, 64, 128, 256, 512, 1024, 2048] with corresponding mel bins [5, 10, 20, 40, 80, 160, 320]. Additionally, we utilize ViSQOL, a perceptually-motivated model for measuring audio

---

[1] https://pytorch.org/audio/main/generated/torchaudio.compliance.kaldi.fbank.html

Table 2: *Comparative Analysis of Reconstruction Performance*

Model (Enc.)	Bitrate	Params	Mel ↓	STFT ↓	ViSQOL ↑
SpeechTokenizer	6.0 kbps	103.7M	0.76	1.58	4.26
HASRD (BestRQ+)	3.1 kbps	99.5M	**0.64**	**1.48**	**4.50**
HASRD (HuBERT)	4.5 kbps	131.7M	0.76	1.57	4.30

quality, to provide a more comprehensive assessment of reconstruction performance. All hyperparameters and analysis choices were selected on the dev-clean subset, while final results are reported on the test-clean subset of LibriSpeech.

4. Results

4.1. Improved Computational Efficiency of DAC

We conducted ablation studies on 100 hours of LibriSpeech to evaluate the impact of our enhanced CNN encoder on reconstruction quality and efficiency using fbank inputs. Unlike previous DAC models trained on 0.38-second audio segments, we found that increasing the window size improves contextual modeling and enhances reconstruction performance, as shown in Table 1. Furthermore, DAC+fbank reduces GPU memory usage by 25%, enabling a $3\times$ increase in batch size compared to raw audio. Inference speed also improves by $2\times$ on CPU, while the total model parameters decrease by 11.2%, as only half of the encoder parameters are used. Furthermore, introducing depthwise separable convolutions exclusively in the residual CNN layers enhances inference speed by $3\times$ compared to the baseline and further reduces parameters by an additional 29% (compared to DAC+fbank), with only a minor degradation in reconstruction quality relative to the DAC baseline. Finally, we compare the reconstruction quality of our proposed representation factorization approach to SpeechTokenizer in Table 2. By increasing the number of RVQ codebooks from 8 to 9, we raise the bitrate to 3.1 kbps. Despite operating at half the bitrate of SpeechTokenizer, our method consistently achieves superior performance, reducing the mel distance from 0.76 to 0.64, STFT loss from 1.58 to 1.48, and improving the ViSQOL score from 4.26 to 4.50. Additionally, we demonstrate the generalizability of our approach to other SSL models, such as HuBERT, achieving reconstruction quality on par with SpeechTokenizer using a configuration of 9 codebooks (8 RVQ + 1 k-means).

4.2. SSL Pre-Training

In this section, we evaluate the impact of enhancing the proposed CNN encoder from Section 4.1 with attention mechanisms and self-supervised pre-training, as shown in Table 3. First, we highlight the limitations of state-of-the-art high-fidelity codecs like DAC by assessing their representations on the ASR task. It can be seen that fbank features achieve an 18.3% relative improvement in WER over DAC's latent encoder representation, and DAC's discretized representation proves ineffective, yielding a WER of 92%. Following the methodology in [24], we first reproduce the BestRQ model and introduce

Table 3: *Comparative analysis of WER/CER results. For HASRD, only the encoder is used for ASR, but total parameters are reported for system-level comparison.*

Model	Params	Norm	CNN Module	Latent		Quantized	
				CER	WER	CER	WER
Fbank	-	-	-	11.8	35.8	-	-
DAC	76.6M	-	1D-CNN	14.6	43.8	52.6	92.6
BestRQ (baseline) [29]	106.0M	global	2D-CNN	4.9	17.3	12.8	38.3
BestRQ+	62.0M	global	1D-CNN+SepConv	3.7	13.2	12.6	37.4
BestRQ+	62.0M	sentence	1D-CNN+SepConv	3.6	12.6	9.3	28.0
SpeechTokenizer	103.7M	-	-	6.5	21.5	7.1	23.0
HASRD (BestRQ+)	141.6M	sentence	1D-CNN+SepConv	3.4	12.0	6.4	21.0
HASRD (HuBERT)	131.7M	-	-	2.1	7.4	3.3	11.3

Table 4: *Speaker similarity and ASR performance using semantic and acoustic tokens*

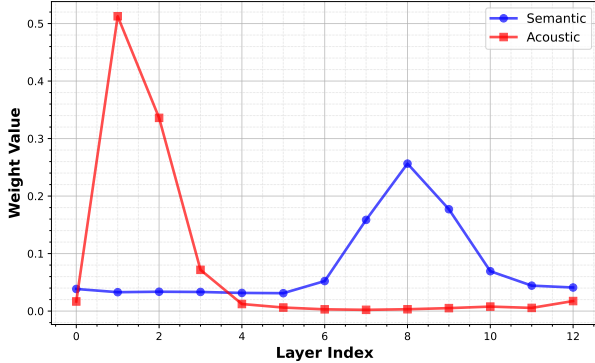| Tokens	SIM ↑	WER ↓
Combined	0.92	24.2
Acoustic	0.67	70.2
Semantic	0.15	21.0

Figure 2: *Weights of latent representations of SSL in acoustic (reconstruction) and semantic (ASR) tasks.*

key improvements. Specifically, we replace the 2D-CNN pre-encoder with our optimized 1D-CNN pre-encoder from Section 4.1, with modifications on the encoder from Section 3. These modifications result in a 23.7% relative WER improvement while reducing the model size to 62M parameters. Additionally, we find that switching from global normalization to sentence normalization reduces feature variance in the latent space from an average of 6 to 1, further improving WER by 4.5% for latent representations and 14.4% for quantized representations with k-means. Increasing the attention dimension from 516 to 768 in HASARD (BestRQ+) yields additional relative WER improvements of 5% for latent representations and 25% for quantized representations compared to BestRQ+, surpassing SpeechTokenizer. Notably, SpeechTokenizer suffers significant WER degradation compared to its teacher model, HuBERT base, due to the distillation process. In contrast, our approach with Hubert, HASRD (HuBERT), effectively retains the performance of pre-trained SSL models.

4.3. Information Disentanglement

In this section, we examine the disentanglement of acoustic and semantic information in HASRD. A key advantage of our method is its interpretability, as the acoustic representation is formed as a convex combination of SSL latent representations, with learned weights as described in Section 2.2. Figure 2 visualizes the weights α learned from the speech reconstruction task and a separate set optimized on the ASR task (Section 3 **Evaluation**), showing the contributions of SSL latent representations to acoustic (reconstruction) and semantic (ASR) objectives. The representations begin with the CNN encoder, corresponding to index 0. The plot reveals that for speech reconstruction, the first three attention layers contribute the most,

with layer indexed 1 receiving the highest weight. Interestingly, the CNN representation has a relatively low weight. We hypothesize that the first attention layer effectively integrates the information encoded by the CNN while capturing contextual dependencies and long-term relationships, reducing the direct contribution of the CNN representation. In contrast, for ASR, the highest weights are observed in layers 7–9, with layer 8 being the most influential, aligning with our ASR experiments. This analysis empirically demonstrates the inherent conflict between acoustic and semantic objectives, where improving one often degrades the other. Additionally, upon listening to the reconstructed speech using only semantic tokens (from the k-means codebook), we noticed that the speech remains intelligible but exhibits a robotic quality, making speaker identification challenging. Conversely, when reconstructing speech using only acoustic tokens (from the remaining RVQ codebooks), speaker identity is well-preserved, but intelligibility is lost. We hypothesize that our disentanglement approach effectively separates semantic and speaker-specific information, with the first codebook capturing linguistic content while the acoustic codebooks encode speaker characteristics. To validate this, we conduct a speaker similarity analysis using the ECAPA-TDNN speaker verification model [31], measuring the cosine similarity between reference and reconstructed speech representations. Additionally, we evaluate ASR performance using both acoustic and semantic tokens to assess their respective contributions, as shown in Table 4. The results show that acoustic tokens achieve a high speaker similarity score (0.67) but poor ASR performance (72% WER), whereas semantic tokens yield lower speaker similarity (0.15) but better ASR performance (21% WER). This confirms that our approach effectively disentangles semantic content (first codebook) from speaker information (acoustic codebooks).

5. Conclusion

In this paper, we proposed a novel framework for learning hierarchical disentangled acoustic and semantic representations. Our approach effectively preserves semantic performance for ASR while achieving reconstruction quality comparable to state-of-the-art neural audio codecs like DAC. Additionally, we introduced an efficient CNN pre-encoder that reduces computational complexity without sacrificing quality, and presented an improved BestRQ design with better ASR performance and enhanced quantized representations. Our analysis provided insights into the inherent conflict between acoustic and semantic objectives, demonstrating that our method successfully disentangles speaker-specific details from linguistic content. Finally, speaker similarity and ASR evaluations further validated the effectiveness of our disentanglement strategy.

6. References

[1] Y. Xu, S.-X. Zhang, J. Yu, Z. Wu, and D. Yu, "Comparing discrete and continuous space LLMs for speech recognition," *arXiv preprint arXiv:2409.00800*, 2024.

[2] Z. Borsos, R. Marinier, D. Vincent, E. Kharitonov, O. Pietquin, M. Sharifi, D. Roblek, O. Teboul, D. Grangier, M. Tagliasacchi, *et al.*, "AudioLM: A language modeling approach to audio generation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 31, pp. 2523–2533, 2023.

[3] J. Peng, Y. Wang, Y. Xi, X. Li, X. Zhang, and K. Yu, "A survey on speech large language models," *arXiv preprint arXiv:2410.18908*, 2024.

[4] D. Zhang, S. Li, X. Zhang, J. Zhan, P. Wang, Y. Zhou, and X. Qiu, "SpeechGPT: Empowering large language models with intrinsic cross-modal conversational abilities," in *Proc. EMNLP*, 2023, pp. 15757–15773.

[5] X. Zhang, D. Zhang, S. Li, Y. Zhou, and X. Qiu, "SpeechTokenizer: Unified speech tokenizer for speech large language models," *arXiv preprint arXiv:2308.16692*, 2023.

[6] X. Chang, B. Yan, K. Choi, J.-W. Jung, Y. Lu, S. Maiti, R. Sharma, J. Shi, J. Tian, S. Watanabe, *et al.*, "Exploring speech recognition, translation, and understanding with discrete speech units: A comparative study," in *Proc. ICASSP*, 2024, pp. 11481–11485.

[7] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "HuBERT: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 29, pp. 3451–3460, 2021.

[8] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao, *et al.*, "WavLM: Large-scale self-supervised pre-training for full stack speech processing," *IEEE J. Sel. Top. Signal Process.*, vol. 16, no. 6, pp. 1505–1518, 2022.

[9] Y.-A. Chung, Y. Zhang, W. Han, C.-C. Chiu, J. Qin, R. Pang, and Y. Wu, "W2v-BERT: Combining contrastive learning and masked language modeling for self-supervised speech pre-training," in *Proc. ASRU*, 2021, pp. 244–250.

[10] X. Chang, B. Yan, Y. Fujita, T. Maekaku, and S. Watanabe, "Exploration of efficient end-to-end ASR using discretized input from self-supervised learning," in *Proc. Interspeech*, vol. 2023, 2023, pp. 1399–1403.

[11] A. Polyak, Y. Adi, J. Copet, E. Kharitonov, K. Lakhotia, W.-N. Hsu, A. Mohamed, and E. Dupoux, "Speech resynthesis from discrete disentangled self-supervised representations," in *Proc. Interspeech*, 2021, pp. 3615–3619.

[12] T. A. Nguyen, W.-N. Hsu, A. d'Avirro, B. Shi, I. Gat, M. Fazel-Zarani, T. Remez, J. Copet, G. Synnaeve, M. Hassid, *et al.*, "Expresso: A benchmark and analysis of discrete expressive speech resynthesis," in *Proc. Interspeech*, 2023, pp. 4823–4827.

[13] N. Zeghidour, A. Luebs, A. Omran, J. Skoglund, and M. Tagliasacchi, "SoundStream: An end-to-end neural audio codec," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 30, pp. 495–507, 2021.

[14] A. Défossez, J. Copet, G. Synnaeve, and Y. Adi, "High fidelity neural audio compression," *arXiv preprint arXiv:2210.13438*, 2022.

[15] A. Défossez, L. Mazaré, M. Orsini, A. Royer, P. Pérez, H. Jégou, E. Grave, and N. Zeghidour, "Moshi: A speech-text foundation model for real-time dialogue," *arXiv preprint arXiv:2410.00037*, 2024.

[16] R. Kumar, P. Seetharaman, A. Luebs, I. Kumar, and K. Kumar, "High-fidelity audio compression with improved rvqgan," in *Proc. NeurIPS*, 2023, pp. 27980–27993.

[17] Z. Ye, P. Sun, J. Lei, H. Lin, X. Tan, Z. Dai, Q. Kong, J. Chen, J. Pan, Q. Liu, *et al.*, "Codec does matter: Exploring the semantic shortcoming of codec for audio language model," *arXiv preprint arXiv:2408.17175*, 2024.

[18] Y. Sim, J. Yoon, and Y.-J. Suh, "SKQVC: One-shot voice conversion by k-means quantization with self-supervised speech representations," *arXiv preprint arXiv:2411.16147*, 2024.

[19] J. Shi, X. Ma, H. Inaguma, A. Sun, and S. Watanabe, "MMM: Multi-layer multi-residual multi-stream discrete speech representation from self-supervised learning model," in *Proc. Interspeech*, 2024, pp. 2569–2573.

[20] Y. Yang, F. Shen, C. Du, Z. Ma, K. Yu, D. Povey, and X. Chen, "Towards universal speech discrete tokens: A case study for ASR and TTS," in *Proc. ICASSP*, 2024, pp. 10401–10405.

[21] W.-N. Hsu, T. Remez, B. Shi, J. Donley, and Y. Adi, "ReVISE: Self-supervised speech resynthesis with visual input for universal and generalized speech regeneration," in *Proc. CVPR*, 2023, pp. 18796–18806.

[22] Z. Mu, X. Yang, S. Sun, and Q. Yang, "Self-supervised disentangled representation learning for robust target speech extraction," in *Proc. AAAI*, vol. 38, 2024, pp. 18815–18823.

[23] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," in *Proc. CVPR*, 2022, pp. 11976–11986.

[24] C.-C. Chiu, J. Qin, Y. Zhang, J. Yu, and Y. Wu, "Self-supervised learning with random-projection quantizer for speech recognition," in *Proc. ICML*, 2022, pp. 3915–3924.

[25] K. Kumar, R. Kumar, T. De Boissiere, L. Gestin, W. Z. Teoh, J. Sotelo, A. De Brebisson, Y. Bengio, and A. C. Courville, "MelGAN: Generative adversarial networks for conditional waveform synthesis," *Proc. NeurIPS*, vol. 32, 2019.

[26] J. Kong, J. Kim, and J. Bae, "HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis," *Proc. NeurIPS*, vol. 33, pp. 17022–17033, 2020.

[27] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "LibriSpeech: An ASR corpus based on public domain audio books," in *Proc. ICASSP*, 2015, pp. 5206–5210.

[28] P. Żelasko, D. Povey, J. Trmal, S. Khudanpur, *et al.*, "Lhotse: A speech data representation library for the modern deep learning ecosystem," in *NeurIPS Data-Centric AI Workshop*, 2021.

[29] R. Whetten, T. Parcollet, M. Dinarelli, and Y. Estève, "Open implementation and study of BEST-RQ for speech processing," *arXiv preprint arXiv:2405.04296*, 2024.

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

[31] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized channel attention, propagation and aggregation in TDNN based speaker verification," in *Proc. Interspeech*, 2020, pp. 3830–3834.