NecoBERT: Self-Supervised Learning Model Trained by Masked Language Modeling on Rich Acoustic Features Derived from Neural Audio Codec

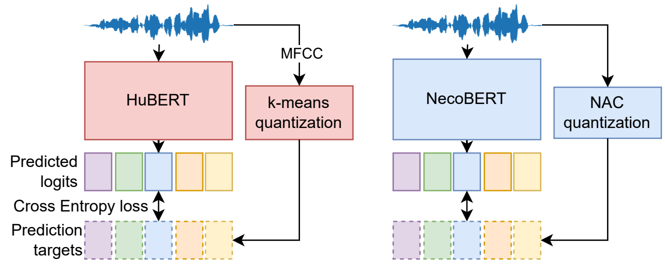
Wataru Nakata* Takaaki Saeki* Yuki Saito* Shinnosuke Takamichi*† and Hiroshi Saruwatari*
* The University of Tokyo, Japan
E-mail: nakata-wataru855@g.ecc.u-tokyo.ac.jp
† Keio University, Japan

***Abstract*—We propose *NecoBERT,* a self-supervised learning (SSL) model to extract speech features containing semantic and acoustic information. Unsupervisedly learned representations using massive speech data, such as SSL and neural audio codec (NAC) features, are essential for developing speech foundation models. However, because conventional methods typically capture one specific aspect of speech, i.e., semantic or acoustic information, the learned features are not always suitable for complicated downstream tasks. NecoBERT addresses this issue by masked language modeling similar to HuBERT on NAC-derived rich acoustic features, enabling the acoustic features to represent semantic information as well. We evaluate NecoBERT on recognition tasks taken from the SUPERB benchmark and speech resynthesis tasks. The results show that although NecoBERT does not outperform HuBERT in some recognition tasks, it performs superiorly in speaker verification and speech resynthesis.**

I. Introduction

Speech pretraining using massive data is essential for the developing speech foundation models, i.e., universal spoken language processing (SLP) models [1]. One mainstream approach is self-supervised learning (SSL) [2]–[4], which trains large deep neural networks (DNNs) as a function to extract speech features useful for various downstream tasks [5]. The core technology is to learn mainly *phonetic or semantic* information from speech [6] during pretraining with context modeling, such as masked language modeling (MLM) [7] and contrastive learning [8]. Another approach is speech reconstruction to extract latent variables related to *acoustic* properties of speech. Neural audio codec (NAC) [9] is a typical example of this approach, which aims to compress input audio using residual vector quantization (RVQ) [10] and adversarial training [11] while losing as little of the fidelity of the original audio as possible.

Following the success in many speech *recognition* tasks covered by the SUPERB benchmark [5], SSL techniques have been transferred to speech *generation* tasks [12]–[14]. For instance, SSL features have recently been used as speech representations for speech synthesis, instead of traditional acoustic features such as mel-spectrograms [15], [16]. Performing k-means clustering [17] of speech features derived from a pretrained SSL model, e.g., HuBERT [3], enables us to define discrete speech tokens that can be used as intermediate representations used for speech synthesis [14]. Although high-fidelity speech synthesis involves modeling of both linguistic and non-/para-linguistic information, previous methods often suffer from poor intelligibility or unnatural prosody and thus result in the low fidelity of generated speech [18], [19]. This issue hinders potential applications of learned speech representations in more practical and complicated SLP tasks such as the modeling of spoken dialogue [20].



(a) HuBERT: MLM on *recognition*-oriented features

(b) NecoBERT: MLM on *reconstruction*-oriented features

Fig. 1: Differences between (a) conventional HuBERT and (b) proposed NecoBERT.

To this end, we propose *NecoBERT* (**Ne**rual audio **co**dec **BERT**), an SSL model aimed at modeling both semantic and acoustic features of speech. Unlike the conventional HuBERT [3] which relies on the results of k-means clustering on mel-frequency cepstrum coefficients (MFCC), NecoBERT is trained to predict speech tokens quantized by a pretrained NAC model, as shown in Figure 1. This training involves MLM on NAC-derived features to contextualize them and enables the learned features to represent semantic information extracted from rich acoustic features. We evaluate NecoBERT in speech processing tasks including recognition tasks [5] as well as speech resynthesis (i.e., neural vocoding and unit-to-speech) to investigate what kind of information the NecoBERT-derived features represent. Our contributions are as follows.

- We propose *NecoBERT*, a novel speech SSL model that can learn both acoustic and semantic features. The training code and the trained model are available online[1].
- From the evaluation results of recognition tasks, we

[1] https://github.com/Wataru-Nakata/SSL4SpeechSynthesis

show that NecoBERT achieves the highest performance in speaker verification compared to NAC and HuBERT.
- From the evaluation results of speech resynthesis tasks, we demonstrate that NecoBERT improves F0 prediction performance and naturalness of resynthesized speech, regardless of whether we quantize the SSL features.

II. Related Work

A. Conventional SSL model: HuBERT

Despite being initially designed for automatic speech recognition (ASR), SSL models have recently demonstrated remarkable performance as feature extractors for various SLP tasks [5]. These models undergo pretraining on abundant speech data initially, which serves as a foundation for subsequent applications across diverse downstream tasks.

HuBERT, for instance, exemplifies this process through two pretraining iterations [3]. Specifically, it first initializes a HuBERT model including Transformer encoders [21] with a convolutional layer by MLM on k-means-clustered MFCCs. Then, it refines the model through another round of MLM on the k-means-clustered 6th layer hidden states of the initial model.

B. NAC and its application in speech synthesis

NAC has recently been established as another approach to discretize audio signals using DNNs trained in an unsupervised manner. The learning process in NAC relies on the use of an information-discretizing bottleneck in an autoencoder structure [9], [22], [23]. By feeding an audio waveform into the encoder part of the trained NAC (i.e., NAC encoder), acoustic tokens descritized by an RVQ layer can be obtained.

As NAC features are expected to contain abundant information related to the acoustic property of original audio, they have been gathering attention as novel representations suitable for speech synthesis [13]. However, NAC faces challenges in capturing temporal dependencies between tokens, and there is no guarantee that the prosody of synthesized speech can be reproduced with high accuracy [19].

III. Proposed SSL Model: NecoBERT

In this work, we propose NecoBERT, which aims to model both semantic and acoustic features from speech. Figure 2 shows the overall model architecture and the training flow of NecoBERT. Our NecoBERT utilizes NAC features suitable for high-fidelity speech reconstruction and learns contextual/semantic information from them by MLM. This strategy is inspired by previous methods that perform MLM on pretrained SSL features, such as vq-wav2vec [2] and w2v-BERT [24].

A. Core architecture

NecoBERT consists of a pretrained NAC encoder and Transformer encoder [21]. The NAC encoder takes raw speech waveform as input and generates d-dimensional NAC latent features $X = \{\boldsymbol{x}_n \in \mathbb{R}^d | n = 1, \cdots, N\}$, where N denotes the number of frames in the feature. The Transformer encoder $f_\theta(\cdot)$ parameterized by θ predicts quantized NAC features (i.e., NAC tokens) $Z = \{\boldsymbol{z}_n \in \mathbb{N} | n = 1, \cdots, N\}$ from X with the positional encoding.

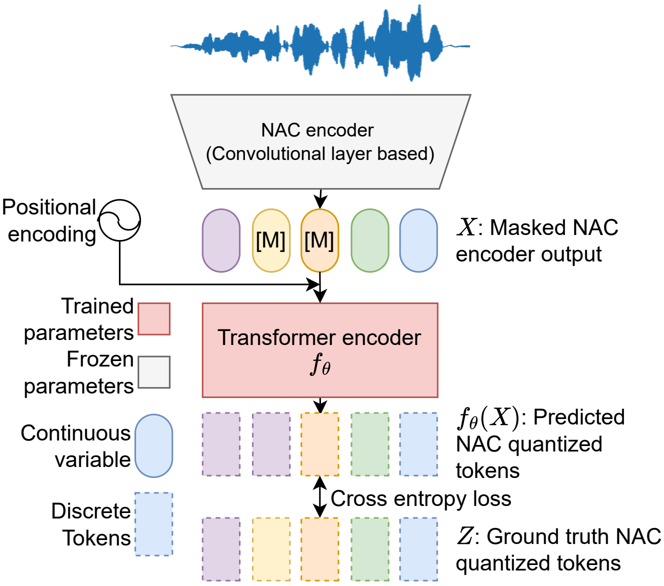


Fig. 2: Model architecture and the training flow of NecoBERT.

Unlike the original framework that uses not only speech but also music and environmental sounds [23], we pretrain the NAC on speech data only. This pretraining enables the NAC-discretized tokens to only capture speech related information, not general audio properties. We empirically observed that this speech-only pretraining significantly improved the speech reconstruction performance of NAC.

B. MLM-based token prediction training

The training of NecoBERT is based on MLM similar to that in existing speech SSL models, which aims to learn contextual information of speech from the NAC-derived latent features. We adopt SpanBERT-based masking [25] following previous work [3]. Specifically, we first select a random segment with a length of l from a target NAC token sequence with a probability of p and mask the selected segment. Then, we replace the masked tokens with a learnable parameter.

C. Training objective and inference

The training loss L is defined as the cross-entropy (CE) between the predicted and target token sequences:

$$L = \sum_{t \in \mathcal{T}} \mathrm{CE}\left(f_\theta(X), Z\right), \qquad (1)$$

where T and $\mathcal{T}$ denote the frame length and the set of frame indices to calculate the loss, respectively. In the previous work [3], calculating loss only on the masked regions was effective in ASR tasks. However, its effectiveness in other tasks remains unclear. To see how the model performance differs depending on the loss calculation strategy, we prepared two variants of NecoBERT. The first one is NecoBERT, which is trained with loss calculated in all regions, i.e., $\mathcal{T} = \{1, \ldots, N\}$. The second one is NecoBERT-masked, which is trained with loss calculated only in masked regions, i.e., $\mathcal{T} = \{m, \dots, m+l\}$ where m denotes the frame to start the l-length masking.

After the training, we can use the last hidden state of the Transformer encoder in NecoBERT as a speech representation. The representation is expected to represent semantic information extracted from rich acoustic features, such as prosody of input speech.

IV. Experiment

In this work, we trained our NecoBERT models and evaluated their performance on recognition tasks and speech resynthesis from continuous SSL features or their discretized versions.

A. Experimental conditions

For the training, we used LibriSpeech [26] containing 960 hours of English multi-speaker speech samples from audiobooks. For NAC, we used Descript Audio Codec (DAC) [23] and trained it on the LibriSpeech corpus for 150k steps. We resampled all speech data to 24 kHz. For DAC parameters, we set the downsample rate of the encoder to $2, 3, 4, 4, 5$, which resulted in downsampling of $1/480$. The output of the DAC was then set to 50 Hz at an input sampling rate of 24 kHz. This output frequency of 50 Hz was the same as those used in HuBERT [3], WavLM [4], and wav2vec 2.0 [2]. For other parameters, we followed the official implementation of DAC[2].

There were 12 Transformer [21] encoder layers in NecoBERT, and their hidden sizes and number of attention heads were set to 768 and 12, respectively. This was the same configuration for wav2vec 2.0, WavLM-base, and HuBERT-base models. For the speech tokens used for the output of NecoBERT (i.e., Z), we only used the output of the first RVQ layer of DAC. This is because pretraining DAC on speech data only enabled the first layer to capture enough acoustic information in speech. Furthermore, in our preliminary experiments, we empirically confirmed that the use of multiple layer quantization results (i.e., multi-stream NAC tokens) made the pretraining unstable. Therefore, we only used output of the first RVQ layer of DAC in this study.

For optimization, we used AdamW [27], setting its parameters as $\beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 1\times10^{-6}$,, and $\lambda = 1\times10^{-2}$. For the learning rate, we linearly increased it from 0 to 2×10^{-4} for first 40k steps and linearly decreased it from 2×10^{-4} to 0 for 460k steps. The total number of training steps is 500k. For MLM, we set the masking probability to $p = 0.08$ and the length of masking to $l = 10$. The training took approximately two days with eight NVIDIA A100 GPUs. NecoBERT consisted of approximately 72M and 91M parameters for the NAC encoder and Transformer encoder, respectively.

B. Evaluation in recognition tasks

We first evaluated our NecoBERT on recognition tasks.

TABLE I: Evaluation results on recognition tasks. Result for HuBERT is quoted from [5]. **Bold** and $\underline{\text{underlined}}$ values are the best and worst in each task, respectively.

Task type	ASR	ASV	ER	SD
Metric	WER↓	EER ↓	ACC ↑	DER ↓
HuBERT	**6.42**	$\underline{5.11}$	**64.92**	**5.88**
DAC	$\underline{29.8}$	2.60	$\underline{48.31}$	$\underline{12.99}$
NecoBERT	21.8	**1.80**	53.57	9.96
NecoBERT-masked	21.7	2.79	57.63	8.36

Task description: For recognition tasks, we used automatic speech recognition (ASR), automatic speaker verification (ASV), emotion recognition (ER), speaker diarization (SD), and intent classification (IC) from SUPERB [5].There tasks correspond to the recognition of content (ASR), speaker (ASV), paralinguistics (ER and SD), and semantic information from speech (IC). We selected these tasks to cover all task types introduced in SUPERB and to avoid high computational costs due to large datasets and diverse tasks [28].

Settings and metrics: For the training condition for each task, we followed the conditions presented in the SUPERB paper [5]. For the compared methods, we used the DAC encoder output and the two proposed models: NecoBERT and NecoBERT-masked described in Section III-C. We kept the parameters of these models unchanged and only used them as feature extractors. For NecoBERT and NecoBERT-masked, we took a weighted sum of the hidden states in the Transformer encoder. For the evaluation metrics, we used word error rate (WER) for ASR, equal error rate (EER) for ASV, classification accuracy (ACC) for ER and IC, and diarization error rate (DER) for SD.

Results: Table I shows the evaluation results on recognition tasks. Note that the result of HuBERT is quoted from the original SUPERB paper [5]. From the results, we can see that the NecoBERT improves all evaluation metrics compared to the DAC. One noteworthy result is that NecoBERT achieves the lowest EER among the compared models including HuBERT, the best-performing model regarding ASV-EER in the SUPERB paper. This result shows that through MLM, NecoBERT successfully captured additional semantic information over DAC. NecoBERT-masked performed better than original NecoBERT in all tasks except for the ASV tasks. One reason is that by calculating loss on only masked regions, the MLM training becomes closer to language modeling and thus the learned features change to contain more semantic information rather than acoustic information [3]. Additionally, HuBERT outperforms NecoBERT in most tasks except for the ASV. This result indicates that the HuBERT is actually trained to capture rich semantic information better than NecoBERT.

C. Evaluation in neural vocoding

We then evaluated NecoBERT on neural vocoding, a task of generating speech using *continuous* speech representations.

Task description: To evaluate the performance on the modeling of acoustic information, such as speaker identity and prosody, we performed neural vocoding from the continuous

[2] https://github.com/descriptinc/descript-audio-codec

TABLE II: Evaluation results on speech resynthesis tasks. **Bold** naturalness MOS values are significantly higher than the baseline results ($p < 0.05$). Underlined values denote the worst-performing ones in objective evaluations.

(a) Results for neural vocoding: resynthesis from continuous speech features

Feature extraction method	LogF0 RMSE ↓	MCD [dB]↓	XVector-sim ↑	Naturalness MOS ↑
Ground-truth	-	-	-	**3.82** ±0.187
Mel-spec.	2.82	2.43	0.987	3.54 ±0.222
HuBERT (baseline)	4.43	5.26	0.953	3.37 ±0.206
NecoBERT	2.71	3.24	0.976	**3.67** ±0.206
NecoBERT-masked	2.73	3.41	0.975	**3.79** ±0.201

(b) Results for u2s: resynthesis from discretized speech features

Feature extraction method	LogF0 RMSE ↓	MCD [dB] ↓	XVector-sim ↑	Naturalness MOS ↑
Ground-truth	-	-	-	**4.06** ±0.18
DAC tokens	4.88	5.34	0.888	**2.75** ±0.19
HuBERT (baseline)	7.70	7.61	0.830	1.94 ±0.17
NecoBERT	4.29	5.65	0.874	**2.78** ±0.21
NecoBERT-masked	6.23	6.52	0.855	1.25 ±0.19

features obtained from the trained SSL models. For this task, we used the HiFi-GAN vocoder [29] and trained it on the train-clean-100, clean-360, other-500 subset of LibriTTS [30]. The evaluation data were taken from the test-clean subset of LibriTTS. We compared mel-spectrogram (Mel-spec.) and the last layer hidden states of HuBERT, NecoBERT, and NecoBERT-masked as the input features of HiFi-GAN. The mel-spectrograms were extracted from the speech with a window length of 1024 and a hop length of 256. We added Mel-spec. to the compared methods because it is widely used for speech representation and as the training target in speech synthesis. We implemented HuBERT using the official weight of HuBERT-base distributed on Github[3]. This HuBERT model was trained on LibriSpeech, the same as NecoBERT and NecoBERT-masked.

Settings and metrics: We evaluated the quality of synthesized speech using objective and subjective metrics. For objective metrics, we used square-root mean squared error of log F0 (LogF0 RMSE), mel-cepstrum distortion (MCD) [31], and x-vector [32] cosine similarity (XVector-sim). These three metrics quantify the prediction performances of prosody, vocal timber, and speaker identity of target speech, respectively. We used the WavLM-base speaker verification model available on huggingface[4] to extract the x-vector from speech. We used WORLD vocoder [33] for the F0 analysis. For subjective metrics, we used naturalness mean opinion score (MOS). For this metric, we employed native English speakers via crowdsourcing using Prolific[5] and asked them to rate how natural the presented speech was on a 5-point scale from 1 (very poor) to 5 (very good). There were 60 raters, and each rater evaluated 10 samples.

Results: Table IIa shows the evaluation results on neural vocoding using continuous SSL features. From the objective evaluation results, Mel-spec. achieved the best MCD and XVector-sim values among the compared methods. However, focusing on the LogF0 RMSE values, our NecoBERT and NecoBERT-masked outperformed the others. This result suggests that the NecoBERT-derived features contain richer prosody information by contextualizing DAC-derived acoustic latent variables. From the subjective evaluation results, there were no statistically significant differences between Ground-truth, Mel-spec., NecoBERT and NecoBERT-masked. This result demonstrates that NecoBERT can achieve naturalness comparable to the ground truth. On the other hand, HuBERT performed the worst in this task and had significantly worse naturalness MOS than Ground-truth, NecoBERT, and NecoBERT-masked. This reveals that the HuBERT-derived features lack acoustic information to reconstruct original speech with high naturalness.

D. Evaluation in unit-to-speech (u2s)

Finally, we evaluated NecoBERT on u2s, a task of generating speech using *discrete* speech representations.

Task description: u2s was originally introduced in [1] and aims to resynthesize speech from its discretized representations. Although the descritization of speech enables us to introduce effective methods widely used in the natural language processing field, it may lose the fine structure of original speech that is required for the speech reconstruction. Therefore, we can regard u2s as a more challenging speech resynthesis task than neural vocoding using continuous SSL features. To evaluate the effectiveness of NecoBERT as a speech discretization method, we conducted an evaluation in u2s. In this task, we first extracted continuous features from the speech using NecoBERT in the same manner as in Section IV-C. The extracted features were then discretized using k-means clustering. The number of clusters was set to 1,000. We trained HiFi-GAN [29], which reconstructs the original speech from the discretized representation, using the same training data as described in Section IV-C. HiFi-GAN contained an embedding layer to convert discrete representation to continuous representation. The evaluation data were the test-clean

[3] https://github.com/facebookresearch/fairseq/blob/main/examples/hubert/README.md

[4] https://huggingface.co/microsoft/wavlm-base-plus-sv

[5] https://www.prolific.com/

subset of the LibriTTS corpus. We compared DAC encoder output (DAC tokens), 6th-layer hidden state of HuBERT, 12th-layer hidden state of NecoBERT, and 12th-layer hidden state of NecoBERT-masked. We used the same settings and metrics as described in Section IV-C. For HuBERT, we selected the 6th-layer hidden state following the previous work [1]. For this task, we added DAC tokens as a compared feature extraction method. This is because, in contrast to DAC tokens, mel-spectrogram is unsuitable for the discretization with k-means clustering.

Results: Table IIb shows the evaluation result on u2s. From objective evaluation results, HuBERT performed the worst among the compared methods, indicating that the HuBERT-derived features are inappropriate for speech resynthesis tasks (i.e., both neural vocoding and u2s). In contrast, DAC tokens performed reasonably well in this task with the best LogF0 RMSE and XVector-sim values. This observation is unsurprising because DAC indeed aims to learn quantized representations of speech with high speech reconstruction performance. Our NecoBERT performed comparably to DAC and had the lowest LogF0 RMSE value. This result suggests that the NecoBERT-derived features tend to capture speech prosody better than existing SSL models and NAC by using MLM on DAC-derived features. One noteworthy point is that NecoBERT-masked performed significantly worse than HuBERT in the subjective evaluation. This observation contradicts the result of neural vocoding, suggesting that the latent space of NecoBERT is more suited for discretization by k-means than that of NecoBERT-masked. In summary, the subjective evaluation results reveal a large gap between Ground-truth and u2s-generated speech samples in terms of their naturalness. Further improvement is required for speech resynthesis from the discrete features.

V. Conclusion

In this work, we proposed a novel speech self-supervised learning (SSL) model called *NecoBERT*. It performs masked language modeling (MLM) on neural audio codec (NAC)-derived features and aims to add contextual information on rich acoustic features. We evaluated NecoBERT on recognition tasks and speech resynthesis. The results showed that the while it performed worse in some recognition tasks, it performed better at speech resynthesis than conventional methods leveraging unsupervised learning with massive speech data. Future work includes developing a better discretization method suitable for our NecoBERT and evaluating our model on more challenging tasks such as text-to-speech.

Acknowledgement

This work was supported by JST Moonshot JPMJMS2011.

References

[1] K. Lakhotia, E. Kharitonov, W.-N. Hsu, *et al.*, "On generative spoken language modeling from raw audio," *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 1336–1354, 2021.

[2] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "Wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Proc. NeurIPS*, Vancouver, Canada, Dec. 2020, pp. 12 449–12 460.

[3] 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 Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[4] S. Chen, C. Wang, Z. 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, pp. 1505–1518, 2021.

[5] S.-w. Yang, P.-H. Chi, Y.-S. Chuang, *et al.*, "SUPERB: Speech Processing Universal PERformance Benchmark," in *Proc. INTERSPEECH*, Brno, Czech Republic, Sep. 2021, pp. 1194–1198.

[6] T. Ashihara, T. Moriya, K. Matsuura, *et al.*, "SpeechGLUE: How well can self-supervised speech models capture linguistic knowledge?" In *Proc. INTERSPEECH*, Ireland, Dublin, Aug. 2023, pp. 2888–2892.

[7] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL-HLT*, Minneapolis, U.S.A., Jun. 2019, pp. 4171–4186.

[8] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv*, vol. abs/1807.03748, 2018.

[9] 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*, vol. 30, pp. 495–507, Nov. 2021. DOI: 10.1109/TASLP.2021.3129994.

[10] A. Vasuki and P. Vanathi, "A review of vector quantization techniques," *IEEE Potentials*, vol. 25, no. 4, pp. 39–47, Jul. 2006.

[11] A. Biswas and D. Jia, "Audio codec enhancement with generative adversarial networks," in *Proc. ICASSP*, Barcerona, Spain, May 2020, pp. 356–360.

[12] W.-C. Huang, Y.-C. Wu, T. Hayashi, and T. Toda, "Any-to-one sequence-to-sequence voice conversion using self-supervised discrete speech representations," *ICASSP*, pp. 5944–5948, 2020.

[13] Z. Borsos, R. Marinier, D. Vincent, *et al.*, "AudioLM: A language modeling approach to audio generation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 2523–2533, 2022.

[14] C. Wang, S. Chen, Y. Wu, *et al.*, "Neural codec language models are zero-shot text to speech synthesizers," *arXiv*, vol. abs/2301.02111, 2023.

[15] J. Shen, R. Pang, R. J. Weiss, *et al.*, "Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions," in *Proc. ICASSP*, Calgary, Canada, Apr. 2018, pp. 4779–4783.

[16] N. Li, S. Liu, Y. Liu, S. Zhao, and M. Liu, "Neural speech synthesis with transformer network," in *Proc. AAAI*, Honolulu, U.S.A., Jul. 2019, pp. 6706–6713.

[17] J. A. Hartigan and M. A. Wong, "A k-means clustering algorithm," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979.

[18] Z.-C. Liu, Z.-H. Ling, Y.-J. Hu, J. Pan, J.-W. Wang, and Y.-D. Wu, "Speech synthesis with self-supervisedly learnt prosodic representations," in *Proc. INTERSPEECH*, Dublin, Ireland, Aug. 2023, pp. 7–11.

[19] C. Du, Y. Guo, H. Wang, *et al.*, "VALL-T: Decoder-only generative Transducer for robust and decoding-controllable text-to-speech," *arXiv*, vol. abs/2401.14321, 2024.

[20] T. A. Nguyen, E. Kharitonov, J. Copet, *et al.*, "Generative spoken dialogue language modeling," *Transactions of the Association for Computational Linguistics*, vol. 11, pp. 250–266, 2023.

[21] A. Vaswani, N. M. Shazeer, N. Parmar, *et al.*, "Attention is all you need," in *Proc. NIPS*, 2017.

[22] A. Défossez, J. Copet, G. Synnaeve, and Y. Adi, "High fidelity neural audio compression," *Transactions on Machine Learning Research*, 2023, ISSN: 2835-8856.

[23] R. Kumar, P. Seetharaman, A. Luebs, I. Kumar, and K. Kumar, "High-fidelity audio compression with improved RVQGAN," in *Proc. NeurIPS*, 2023.

[24] Y.-A. Chung, Y. Zhang, W. Han, *et al.*, "w2v-BERT: Combining contrastive learning and masked language modeling for self-supervised speech pre-training," in *Proc. ASRU*, 2021, pp. 244–250.

[25] M. Joshi, D. Chen, Y. Liu, D. S. Weld, L. Zettlemoyer, and O. Levy, "SpanBERT: Improving pre-training by representing and predicting spans," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 64–77, 2019.

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

[27] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proc. ICLR*, 2017.

[28] Y.-H. Wang, H.-Y. Chen, K.-W. Chang, W. Hsu, and H.-y. Lee, "MiniSUPERB: Lightweight benchmark for self-supervised speech models," in *Proc. ASRU*, Taipei, Taiwan, Dec. 2023.

[29] J. Kong, J. Kim, and J. Bae, "HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33, Curran Associates, Inc., 2020, pp. 17 022–17 033.

[30] H. Zen, V. Dang, R. Clark, *et al.*, "LibriTTS: A Corpus Derived from LibriSpeech for Text-to-Speech," in *Proc. Interspeech 2019*, 2019, pp. 1526–1530. DOI: 10.21437/Interspeech.2019-2441.

[31] R. Kubichek, "Mel-cepstral distance measure for objective speech quality assessment," in *Proc. PACRIM*, 1993, pp. 125–128.

[32] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, *X-Vectors: Robust DNN embeddings for speaker recognition*, 2018.

[33] M. Morise, F. Yokomori, and K. Ozawa, "WORLD: A vocoder-based high-quality speech synthesis system for real-time applications," *IEICE transactions on information and systems*, vol. E99-D, no. 7, pp. 1877–1884, 2016.