Using Random Codebooks for Audio Neural AutoEncoders

Benoît Giniès, Xiaoyu Bie, Olivier Fercoq, Gaël Richard
LTCI, Télécom Paris, Institut polytechnique de Paris, Palaiseau, France
E-mail: *firstname.lastname@telecom-paris.fr*

***Abstract*—Latent representation learning has been an active field of study for decades in numerous applications. Inspired among others by the tokenization from Natural Language Processing and motivated by the research of a simple data representation, recent works have introduced a quantization step into the feature extraction. In this work, we propose a novel strategy to build the neural discrete representation by means of random codebooks. These codebooks are obtained by randomly sampling a large, predefined fixed codebook. We experimentally show the merits and potential of our approach in a task of audio compression and reconstruction.**

***Index Terms*—feature extraction, quantization, random codebooks, audio reconstruction**

I. Introduction

The extraction of a meaningful and compact representation of the input data is an essential step of modern machine learning based systems. This representation is expected to extract relevant information about the input data to ease the resolution of a target task such as audio classification [1], speech enhancement [2] or audio inpainting [3]. Numerous architectures have been proposed to obtain such a representation for audio signals in a supervised or unsupervised manner [4], [5]. The objective is usually to express the input data under the form of a continuous latent representation which is then further processed by subsequent blocks for the downstream task. For example, Variational Auto Encoders (VAE) [6], [7] are very popular models which allow to fit a probabilistic distribution to input data, and randomly sample a corresponding latent representation from it.

Traditionally, the goal has been to extract a continuous representation from the input data, but there is nowadays a strong trend towards obtaining a discrete representation which has many advantages for data modeling, prediction or generation. For instance, in the VQ-VAE model introduced in [8], the latent representation, learned in an unsupervised manner, is quantized using vector quantization exploiting a so-called dictionary (or *codebook*) of tokens (or *codewords*). Many variations of this model exploiting multiple subdictionnaries were then introduced either in a global hierarchical multiresolution quantization scheme [9]–[11] or in a successive residual quantizations framework [12], [13]. The latter model is particularly efficient for audio coding and demonstrated excellent performances in sound generation by incorporating an autoregressive language model.

Nevertheless, discrete neural approaches in general face several challenges: first, they are subject to a suboptimal exploitation of the codebooks, the so-called *codebook collapse* problem and they usually rely on ad-hoc heuristics to mitigate this collapse problem; second, there is no clear evidence that all codebooks need to be explicitly learned; and third despite their inherent advantage compared to approaches based on a continuous latent representation, they may still suffer from limited generalization capabilities.

In this work, we propose an alternative strategy to build the discrete latent representation. Inspired by the work of Moussallam et al. on Sequential Subdictionaries Matching Pursuits [14], the core idea of our method is to obtain successive small dictionaries randomly sampled from a large fixed dictionary.

Using the recent Descript Audio Codec (DAC) model [15] as a strong baseline, our preliminary experiments show that this novel strategy for codebook design :

- is robust to the codebook collapse problem,
- has potential to avoid the learning of some of the codebooks, leading to a gain of complexity while maintaining compression efficiency and audio reconstruction quality.

The paper is organised as follows: we first describe the related work (section II), before describing in detail our approach (section III). We then present our experimental plan in section IV and discuss the results obtained in sections V and VI. Finally, we highlight some future work and suggest some conclusions in section VII.

II. Related work

A. VAE for audio neural modeling

The objective of the VAE [6], [7] is to fit a probabilistic distribution to input data, by jointly learning the latent generative modeling and the variational posterior distribution. Whereas the original VAEs were primarily applied to image data, subsequent studies have expanded their use to audio data modeling [16] and showcased their effectiveness for various downstream tasks such as speech enhancement [17], voice conversion [18] and text-to-speech generation [19]. Meanwhile, the conventional VAE model relies on single latent space with the *i.i.d* assumption, which motivates researchers to investigate hierarchical [20] and disentangled [21] latent space, and dynamical modeling [22].

This work was funded by the European Union (ERC, HI-Audio, 101052978). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

B. Feature quantization and VQ-VAE

Besides learning continuous representation for audio data, recent studies show the interest of learning discrete, or quantized, representations [23]. While forcing the model to ignore the irrelevant information during quantization, the discrete representations can then be related to high level concepts easily understandable by humans such as phonemes in speech or notes in music. Injecting the feature quantization to a VAE leads to the well-known VQ-VAE model [8], that has been used to generate images [9] and audio [24]. More recently, the residual VQ-VAE [12], [13] has been introduced for audio coding and modeling, which also shows impressive performance on audio generation [25].

C. Codebook collapse

Despite the promising results in many tasks of generating complex data, the standard training approach frequently encounters codebook collapse, *i.e.* only a portion of available codes are actually utilized, largely restricting the quality of the discrete latent representations. To mitigate this problem, many techniques have been proposed, such as exponential moving average (EMA) for codebook update [8], codebook reset [10], a stochastic variant (SQ-VAE) [26], and factorizing the codes with L2-normalization [15].

III. Random Residual Vector Quantization

Our main goal in this work is to explore an alternative strategy for defining, using and training the codebooks used in residual vector quantization (RVQ). As discussed above, dictionary learning is an active area of research, since the training procedures proposed in [8] and used in [13] trigger codebook collapse. Furthermore, given the generative potential of such a quantized representation [23], [25], there is a clear interest to build expressive and general purpose dictionaries.

A straightforward modification along these lines would be to increase the size of the quantization codebooks. However, this would greatly increase the computation complexity (at inference and training) and the bitrates (leading to less efficient audio codecs), and would not solve any codebook collapse issue.

We rather propose an alternative strategy inspired by the Sequential Subdictionaries Matching Pursuits algorithm introduced in [14]. In this algorithm, a signal is reconstructed by a sum of tokens that are iteratively selected from a sequence of dictionaries. The randomness comes from the fact that, at each step, the best token for reconstruction is selected from a small sub-dictionnary built as a random subsampling of a much larger dictionary. Experiments have shown in the context of Matching Pursuit that by proceeding this way, the quality of the reconstruction is almost equivalent to that obtained by using the whole dictionary, for a much lower complexity.

Furthermore, the observations made in [11], as well as the observations we made on DAC model [15], hint that, in a hierarchy of quantizers, deeper quantizers only bring fine grain information to the reconstruction (similar to encoding noise). Thus, it seems relevant to apply the random sampling procedure only to deeper quantizers, as they encode less specific information. In our case, we then obtain these high level quantizers by randomly subsampling a significantly larger codebook, while guaranteeing non-redundant sub-sampling for each quantizer. This permits to directly account for the similar nature of tokens at high level and contribute to enforce generalizability.

As the big codebook is supposed to be large, compared to the initial size of the trained codebooks, we advocate that it does not need to be trained, which is an important advantage for controlling complexity. To further detail our approach, let's say we aim to quantize $x \in \mathbb{R}^D$ using the codebook $\mathcal{B} = (\beta_i)_{i \leq N} \in (\mathbb{R}^D)^N$. The token is selected following $\arg\min_{\beta \in \mathcal{B}} ||x - \beta||_2^2$. Let's now define $\mathcal{B}_{big} \in (\mathbb{R}^D)^{N_{big}}$ a big codebook, instead of computing $\arg\min_{\beta \in \mathcal{B}_{big}} ||x - \beta||_2^2$, we will randomly extract $\mathcal{B}_s \subset \mathcal{B}_{big}$ of size s, and compute $\arg\min_{\beta_s \in \mathcal{B}_s} ||x - \beta_s||_2^2$. A schematic description of that process is introduced in Algorithm 1 and in Figure 1.

Algorithm 1 Random RVQ model

Input: Input signal $x \in \mathbb{R}^D$
Trainable codebooks $(\mathcal{B}_i)_{i \leq n_t}$ sampled from $\mathcal{N}(0, \mathcal{I}_D, N_t)$
Big fixed codebook $\mathcal{B}_{big}$ sampled from $\mathcal{N}(0, \mathcal{I}_D, N_{big})$
Sampling size s
Number of random quantizers n_r
Output: Quantization tokens $(\bar{\beta}_i)_{i \leq n_t + n_r}$
Process:
$residual \leftarrow x$
for $i \leq n_t + n_r$ **do**
  if $i \leq n_t$ **then**
    $\bar{\beta}_i \leftarrow$ NearestNeighbour($residual$, $\mathcal{B}_i$)
  else
    $\mathcal{B}_s \leftarrow$ UniformRandomSampling($\mathcal{B}_{big}$, s)
    $\bar{\beta}_i \leftarrow$ NearestNeighbour($residual$, $\mathcal{B}_s$)
  end if
  $residual \leftarrow residual - \bar{\beta}_i$
end for
return $(\bar{\beta}_i)_{i \leq n_t + n_r}$

IV. Experiments

1) Architecture: To evaluate our approach, we select the same use case (i.e. *audio reconstruction*) as the EnCodec [13] and DAC [15] models. The latter architecture is currently the state of the art for audio compression and reconstruction. Just as for EnCodec, the DAC model is formed of an encoder module, a quantization module and a decoder module. The encoder module extracts continuous 2-D latent features from the input 1-D audio data, then the quantization module discretizes the latent representation and feeds the result to the decoder module which builds a reconstruction of the input signal. The encoder and decoder modules are stacks of convolutional and down/up sampling layers, the quantization module is shaped as a residual vector quantizer.

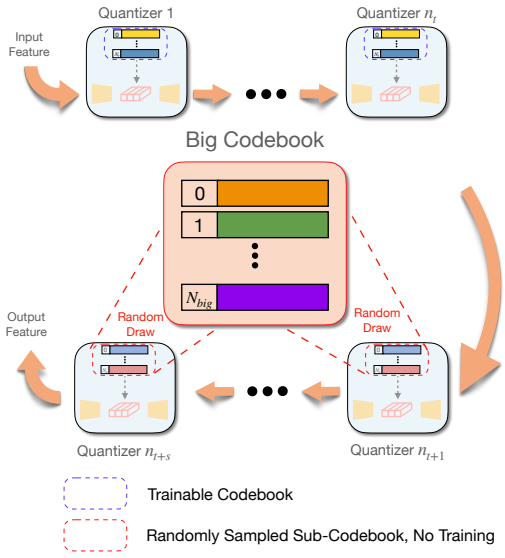

Fig. 1. The illustration of Random RVQ. Same as DAC [15], it contains nine layers of quantizer. However, the first five layers are trainable and the last four layers are randomly sampled from a large codebook, while all layers are trainable in DAC. The dimension reduction (orange rectangles in each quantizer) represents the codebook collapse mitigant operations.

The quantization module is formed of 9 quantizers, which are organized in a pile-like structure, each quantizer computing a new residual after quantization, and feeding it to the next quantizer. Our Random RVQ approach is potentially applicable to all quantizers, but in this work we limit our study to the replacement of the 4 to 6 last quantizers, which given their position in the quantization pile, mostly encode noise.

2) Data: The DAC model [15] is initially trained on mixed audio data: music, speech and environmental sounds extracted from 8 underlying datasets. In this work, to evaluate the potential of our approach, we trained our model on only one of these 8 datasets: MUSDB18 [27]. It is composed of a train set of 100 music tracks, and a test set of 50 music tracks, all sampled at 44.1 kHz with a total length around 10 hours. To obtain a meaningful baseline, we retrained the DAC model on that dataset only. It can be noted that only a very limited decrease of performance was observed after re-training which confirms that the DAC model can be considered as a strong baseline.

3) Evaluation: We kept the same losses as in DAC [15] for the training of the randomized model: generative and feature matching losses (adversarial), a multi-scale mel loss and codebook and commitment losses. We keep the same balancing between the losses as introduced in DAC [15].

To evaluate a model, we also applied the same metrics as the one used for the training of DAC: waveform loss, stft loss, mel loss, scale invariant signal-to-distortion ratio (SI-SDR) as introduced in [28], and the ViSQOL value, a perceptual audio quality assessment introduced in [29].

To monitor the usage of the codebooks, and therefore assess the potential codebook collapse, we report the perplexity of each codebook. The perplexity of a codebook $\mathcal{B}$, composed of N tokens $(\beta_i)_{i=1\dots N}$, is defined as:

$$PP(\mathcal{B}) = e^{-(\sum_{i=1}^{N} \frac{1}{n_i} \ln(\frac{1}{n_i}))}$$

where $(n_i)_{i=1\dots N}$ represent the occurences of tokens used for a given test set. Such a metric converges towards N when the codebook is equally partitioned for quantization. In the case of codebook collapse, the perplexity takes low values.

We also measure the training time, as a statistical estimator for complexity.

4) Collapse mitigants: The initial version of DAC included codebook collapse mitigants, which were compatible with the initial VQ-VAE training trick: the straight through operator trick. These collapse mitigants are the normalization of codebooks and latents before quantization, and the projection of latents on a smaller dimensional space before quantization (the representation is projected back onto the initial space after quantization). The latter mitigant is performed thanks to convolutional layers placed at the input and the ouptut of each quantizer (see Fig. 1).
We discuss in our experiments the impact of removing these mitigants for our model and the baseline.

5) Model variants: We propose five variations of our model to explore the impact of each parameter. The different parameters values characterizing each variant (from RandRVQ1 to RandRVQ5) are displayed in the upper part of Table II. Through these experiments, a variation in the ratio of the size of the big codebook and the random sampling was introduced. We also explored the impact of the collapse mitigants, and the number of random quantizers.

TABLE I
RECONSTRUCTION WITHOUT (LEFT) AND WITH (RIGHT) RANDOMIZED QUANTIZERS

	Baseline 5q	RandRVQ1
mel loss (↓)	0.81	**0.72**
stft loss (↓)	2.10	**2.00**
waveform loss (↓)	0.05	**0.04**
SI-SDR (↑)	6.53	**8.25**
ViSQOL (↑)	3.87	**3.94**

V. RESULTS

The results[1] displayed in Table I show the reconstruction metrics obtained with a partial Baseline (5 trained quantizers) and a total RandRVQ1 (5 trained and 4 randomized quantizers). It clearly shows that even using random codebooks without training, RandRVQ1 can still improve the reconstruction quality compared to a partial Baseline, underlining that deeper quantizers do encode useful information and that it is well captured by our random scheme.

As shown in Table II, the results obtained with our RandRVQ models are, overall, slightly below those of the Baseline.

[1]Some examples are displayed at: https://randrvq.github.io/

TABLE II
OBJECTIVE EVALUATION ON MUSDB18 DATASET (AVERAGED OVER 5 RANDOM DRAWS)

	Baseline	Collapse	RandRVQ1	RandRVQ2	RandRVQ3	RandRVQ4	RandRVQ5
N_{big}	$\emptyset$	$\emptyset$	8192	8192	8192	16384	16384
sample size	$\emptyset$	$\emptyset$	1024	1024	256	512	512
Collapse miti.	$\surd$	$\emptyset$	$\surd$	$\emptyset$	$\emptyset$	$\surd$	$\surd$
# rand. quantizers	0	0	4	4	4	4	6
mel loss (↓)	0.71	0.86	**0.72±0.001**	0.78±0.001	0.79±0.001	0.72±0.001	0.75±0.002
stft loss (↓)	1.99	2.16	**2.00±0.002**	2.07±0.001	2.08±0.001	2.00±0.002	2.03±0.003
waveform loss (↓)	0.041	0.057	**0.042±0.000**	0.048±0.000	0.048±0.000	0.043±0.0001	0.044±0.0001
SI-SDR (↑)	8.40	5.26	**8.25±0.02**	6.99±0.01	7.00±0.01	8.14±0.02	7.76±0.03
ViSQOL (↑)	3.92	3.85	**3.94±0.007**	3.87±0.006	3.88±0.004	3.93±0.008	3.92±0.01
training time (days)	1.55	1.583	1.459	1.465	**1.442**	1.462	1.51

TABLE III
CODEBOOK PERPLEXITIES (PP) ON MUSDB18 DATASET (*in parentheses: ratio to the maximum, trained codebooks are of size 1024*)

	Baseline	Collapse	RandRVQ1	RandRVQ2	RandRVQ3	RandRVQ4	RandRVQ5
PP - cb 1	545 (0.53)	25 (0.02)	569 (0.55)	536 (0.52)	534 (0.52)	545 (0.53)	507 (0.49)
PP - cb 2	834 (0.82)	42 (0.04)	810 (0.79)	810 (0.79)	817 (0.79)	807 (0.78)	828 (0.80)
PP - cb 3	889 (0.86)	67 (0.06)	914 (0.89)	873 (0.85)	905 (0.88)	896 (0.87)	906 (0.88)
PP - cb 4	928 (0.90)	83 (0.08)	913 (0.89)	905 (0.88)	924 (0.90)	919 (0.89)	*15565 (0.95)*
PP - cb 5	935 (0.93)	113 (0.11)	947 (0.92)	950 (0.92)	927 (0.90)	934 (0.91)	*15586 (0.95)*
PP - cb 6	946 (0.92)	140 (0.13)	*7744 (0.94)*	*574 (0.07)*	*3703 (0.45)*	*15601 (0.95)*	*15784 (0.96)*
PP - cb 7	957 (0.93)	137 (0.13)	*7720 (0.94)*	*594 (0.07)*	*3850 (0.46)*	*15659 (0.95)*	*15723 (0.95)*
PP - cb 8	959 (0.93)	166 (0.16)	*7730 (0.94)*	*616 (0.07)*	*3997 (0.48)*	*15622 (0.95)*	*15923 (0.97)*
PP - cb 9	966 (0.94)	145 (0.14)	*7776 (0.94)*	*633 (0.07)*	*4131 (0.50)*	*15730 (0.96)*	*15987 (0.97)*
PP - Big cb	$\emptyset$	$\emptyset$	*7981 (0.97)*	*606 (0.07)*	*3937 (0.48)*	*16090 (0.98)*	*16230 (0.99)*
N_{big}	$\emptyset$	$\emptyset$	*8192*	*8192*	*8192*	*16384*	*16384*
s	$\emptyset$	$\emptyset$	*1024*	*1024*	*256*	*512*	*512*

Yet, except for the SI-SDR metrics of RandRVQ2 and RandRVQ3, the figures are comparable, and even slightly better for VISQOL in RandRVQ1, RandRVQ4 and RandRVQ5. These observations, even if they do not constitute a breakthrough in terms of quality of reconstruction, are positive. Indeed, it indicates that in a setting where we forced the quantization to be untrained, and where the reconstruction is highly dependant on randomness, the obtained quality of the encoding we get from quantization at the same bitrate is globally comparable.

On the complexity side, as expected, our approaches demonstrate a slight advantage with a gain of a couple hours in training compared to the baseline. Nevertheless, this gain remains small.

The codebook perplexities of the different models are given in Table III. As a control experiment, we verify the mitigating effects of the collapse mitigants introduced in DAC [15], as the collapse experiment shows a clear codebook collapse pattern in the perplexities, compared to the Baseline.

Looking at our experiments, we also notice that, overall, the behaviour of the learned first codebooks is similar to the Baseline, meaning that the usage of randomized quantizers does not influence these. Similarly, the perplexities measured for randomized codebooks which are featured with the collapse mitigants of [15], indicate a nearly optimal usage of codebooks. This was predictable, as we are still profiting from the effect of the collapse mitigant, and as the quantization is subject to a random sampling which forces the exploration of the whole codebook.

The results from RandRVQ2 and RandRVQ3 bear more interesting information, as we can clearly see that in the case of no collapse mitigants, and poor parameters selection (RandRVQ2), the randomized quantizers fall back into codebook collapse in spite of the random sampling. This can be (at least partially) solved, by making a better choice of parameters, as we can see that RandRVQ3 has much better values of perplexities over its randomized quantizers (though not perfect, which indicates a possibility to probably go further).

The difference between those two experiments comes from the sampling size that is applied for each random quantizer: the smaller the sampling, the smaller the choice of tokens for the quantizer, which forces exploration of the codebook, but limits the precision of the quantization. Then, the randomization of quantizers, associated with a correct choice of parameters, can be used as a collapse mitigant. Overall, these results are promising and indicate that the randomization of codebooks is potentially relevant for quantization since it can lead to good reconstruction metrics.

VI. Discussion

Although it is shown in this preliminary work that the randomization of quantizers bears promising prospects, some aspects must be further explored to better substantiate the potential of our approach.

1) Fixed codebooks at inference: The "surprising" part of our method is that during inference, the random quantizers are still subject to random sampling. Such randomness could have great implications in terms of generalization, and it would be interesting to study the potential gain in exploiting several random draws to find an optimal quantization. Further explorations could be dedicated to finding a way to fix the sampling of the big codebook at inference.

2) Generalizability: Preliminary experiments evaluating the generalization capabilities of our approach on unseen data (environmental audio data from ESC50) have shown a slightly better robustness than the baseline (although statistically insignificant). Future work is needed to fully explore the robustness potential of our approach.

3) Training the big codebook: Allowing the big codebook to be trained could further improve the reconstruction quality of our model, as the big codebook would still be of a much bigger size than traditional codebooks, and as it would converge through the training to a version of itself that would be adapted to the input data. It is also expected that codebook training may be necessary to allow an extension of the random process to the first codebooks which are apparently capturing more structured information departing clearly from Gaussian noise.

VII. Conclusion

Overall, though still in progress, this exploration has clearly shown that this novel concept of randomization of quantizers, in a context of quantized feature extraction, is very promising. As discussed above, there are several interesting directions that deserve to be pursued to better characterize and substantiate the potential of using random dictionaries including the design of optimal fixed dictionaries at inference, on the generalisation properties and on the extension of the random process to all codebooks of the Residual Vector quantization scheme.

References

[1] G. Richard, S. Sundaram, and S. Narayanan, "An overview on perceptually motivated audio indexing and classification," *Proc. IEEE*, vol. 101, no. 9, pp. 1939–1954, 2013.

[2] Y.-C. Wang, S. Venkataramani, and P. Smaragdis, "Self-supervised learning for speech enhancement," *arXiv preprint arXiv:2006.10388*, 2020.

[3] A. Adler, V. Emiya, M. G. Jafari, M. Elad, R. Gribonval, and M. D. Plumbley, "Audio inpainting," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 3, pp. 922–932, 2012.

[4] A. Mohamed, H.-y. Lee, L. Borgholt, J. D. Havtorn, J. Edin, C. Igel, K. Kirchhoff, S.-W. Li, K. Livescu, L. Maaløe, T. N. Sainath, and S. Watanabe, "Self-supervised speech representation learning: A review," *IEEE J. Sel. Topics Signal Process.*, vol. 16, no. 6, pp. 1179–1210, 2022.

[5] S. Parekh, S. Essid, A. Ozerov, N. Q. Duong, P. Pérez, and G. Richard, "Weakly supervised representation learning for audio-visual scene analysis," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 416–428, 2019.

[6] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," *Proc. Int. Conf. Learn. Repres. (ICLR)*, 2014.

[7] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," *Proc. Int. Conf. Mach. Learn. (ICML)*, 2014.

[8] A. Van Den Oord, O. Vinyals *et al.*, "Neural discrete representation learning," *Advances Neural Inform. Process. Systems (NeurIPS)*, 2017.

[9] A. Razavi, A. Van den Oord, and O. Vinyals, "Generating diverse high-fidelity images with vq-vae-2," *Advances Neural Inform. Process. Systems (NeurIPS)*, 2019.

[10] W. Williams, S. Ringer, T. Ash, D. MacLeod, J. Dougherty, and J. Hughes, "Hierarchical quantized autoencoders," *Advances Neural Inform. Process. Systems (NeurIPS)*, 2020.

[11] Y. Takida, Y. Ikemiya, T. Shibuya, K. Shimada, W. Choi, C.-H. Lai, N. Murata, T. Uesaka, K. Uchida, W.-H. Liao *et al.*, "Hq-vae: Hierarchical discrete representation learning with variational bayes," *arXiv preprint arXiv:2401.00365*, 2023.

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

[13] A. Défossez, J. Copet, G. Synnaeve, and Y. Adi, "High fidelity neural audio compression," *Trans. Mach. Learn. Res.*, 2023.

[14] M. Moussallam, L. Daudet, and G. Richard, "Matching pursuits with random sequential subdictionaries," *Signal Process.*, vol. 92, no. 10, pp. 2532–2544, 2012.

[15] R. Kumar, P. Seetharaman, A. Luebs, I. Kumar, and K. Kumar, "High-fidelity audio compression with improved rvqgan," *Advances Neural Inform. Process. Systems (NeurIPS)*, 2024.

[16] M. Blaauw and J. Bonada, "Modeling and transforming speech using variational autoencoders," *Proc. Interspeech Conf.*, 2016.

[17] X. Bie, S. Leglaive, X. Alameda-Pineda, and L. Girin, "Unsupervised speech enhancement using dynamical variational autoencoders," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 30, pp. 2993–3007, 2022.

[18] J. Lian, C. Zhang, and D. Yu, "Robust disentangled variational speech representation learning for zero-shot voice conversion," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2022.

[19] J. Kim, J. Kong, and J. Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2021.

[20] A. Vahdat and J. Kautz, "Nvae: A deep hierarchical variational autoencoder," *Advances Neural Inform. Process. Systems (NeurIPS)*, 2020.

[21] W.-N. Hsu, Y. Zhang, and J. Glass, "Unsupervised learning of disentangled and interpretable representations from sequential data," in *Advances Neural Inform. Process. Systems (NeurIPS)*, 2017.

[22] X. Bie, L. Girin, S. Leglaive, T. Hueber, and X. Alameda-Pineda, "A benchmark of dynamical variational autoencoders applied to speech spectrogram modeling," *Proc. Interspeech Conf.*, 2021.

[23] K. Lakhotia, E. Kharitonov, W.-N. Hsu, Y. Adi, A. Polyak, B. Bolte, T.-A. Nguyen, J. Copet, A. Baevski, A. Mohamed *et al.*, "On generative spoken language modeling from raw audio," *Trans. Assoc. Comput. Linguistics*, vol. 9, pp. 1336–1354, 2021.

[24] P. Dhariwal, H. Jun, C. Payne, J. W. Kim, A. Radford, and I. Sutskever, "Jukebox: A generative model for music," *arXiv preprint arXiv:2005.00341*, 2020.

[25] 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 Trans. Audio, Speech, Lang. Process.*, vol. 31, pp. 2523–2533, 2023.

[26] Y. Takida, T. Shibuya, W. Liao, C.-H. Lai, J. Ohmura, T. Uesaka, N. Murata, S. Takahashi, T. Kumakura, and Y. Mitsufuji, "Sq-vae: Variational bayes on discrete representation with self-annealed stochastic quantization," *Proc. Int. Conf. Mach. Learn. (ICML)*, 2022.

[27] Z. Rafii, A. Liutkus, F.-R. Stöter, S. I. Mimilakis, and R. Bittner, "The MUSDB18 corpus for music separation," Dec. 2017. [Online]. Available: https://doi.org/10.5281/zenodo.1117372

[28] J. Le Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "Sdr–half-baked or well done?" in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2019.

[29] M. Chinen, F. S. Lim, J. Skoglund, N. Gureev, F. O'Gorman, and A. Hines, "Visqol v3: An open source production ready objective speech and audio metric," in *Proc. Int. Conf. Quality Multimedia Experience (QoMEX)*, 2020.