



NAST: Noise Aware Speech Tokenization for Speech Language Models

Shoval Messica, Yossi Adi

School of Computer Science and Engineering
The Hebrew University of Jerusalem, Israel

shoval.messica@mail.huji.ac.il

Abstract

Speech tokenization is the task of representing speech signals as a sequence of discrete units. Such representations can be later used for various downstream tasks including automatic speech recognition, text-to-speech, etc. More relevant to this study, such representation serves as the basis of *Speech Language Models*. In this work, we tackle the task of speech tokenization under the noisy setup and present **NAST: Noise Aware Speech Tokenization for *Speech Language Models***. NAST is composed of three main components: (i) a predictor; (ii) a residual encoder; and (iii) a decoder. We evaluate the efficiency of NAST considering several spoken language modeling tasks and show that NAST is superior to the evaluated baselines across all setups. Lastly, we analyze NAST and show its disentanglement properties and robustness to signal variations in the form of noise, reverberation, pitch-shift, and time-stretch. Code and pre-trained models are available at <https://github.com/ShovalMessica/NAST>.

Index Terms: speech tokenization, speech language modeling

1. Introduction

Self-supervised models have shown to be highly effective in extracting meaningful representations from raw speech signals [1, 2, 3, 4]. Recently, the authors in [5] demonstrated that such self-supervised representations can be used under the *Generative Spoken Language Modeling (GSLM)* framework.

The GSLM pipeline typically starts with a self-supervised learning model that extracts continuous speech embeddings. These embeddings are then quantized into a discrete form, often using the k-means algorithm [5, 6, 7]. A speech-language model is subsequently trained on these quantized units, which are finally converted back into raw audio through a unit-based neural vocoder. This framework was shown to be effective in modeling multiple levels of the speech utterance: prosody, content [5, 6, 7, 8], speech compression and enhancement [9, 10, 11], voice and emotion conversion [12, 13], spoken dialogue [14], and speech-to-speech translation [15, 16, 17, 18].

Despite their effectiveness, recent findings highlight a susceptibility of such techniques to acoustic variations that do not affect the linguistic content but greatly modify the output representation [19], hence questioning their robustness and generalization. For instance, performing a time-stretch of less than 10% of the speech utterance yields an edit distance of more than 40%. The authors in [19] proposed an augmentation invariant speech tokenizer together with an objective evaluation metric to track progress in the field. Although providing impressive results, the method proposed by [19] is based on a teacher-student paradigm with k-means being the teacher. Hence, inherits k-means properties and bias.

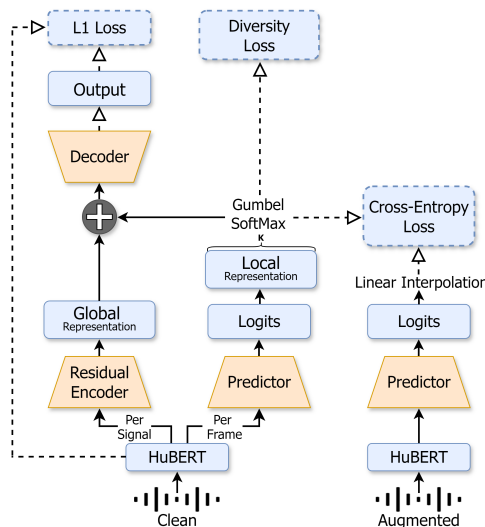


Figure 1: A high-level overview of NAST. Clean and augmented signals are fed into the predictor to produce frame-wise logits. The clean logits undergo Gumbel sampling to become one-hot vectors for local representation. The residual encoder extracts a global representation from the clean signal, merged with local ones for decoder input to reconstruct the original signal embeddings. Augmented signal logits are aligned via linear interpolation for robustness enhancement, and diversity loss is applied over the one-hot vectors to ensure full unit usage.

To confront the above-mentioned issue, in this work, we propose a novel speech tokenization method named **NAST**, which stands for **Noise Aware Speech Tokenization for Speech Language Models**. NAST is composed of three main components: (i) a predictor that maps the speech signal into local discrete representations. Such representation mainly captures local information in the form of phonemes or sub-phonemes; (ii) a residual encoder which predicts a single global representation for the whole sequence. This representation mainly captures global information such as speaker identification; and (iii) a decoder which outputs the original signal representation given both local and global representations. To improve invariance to signal variations we match the representations obtained from the predictor module of both clean and augmented speech signals. All modules are jointly optimized using several loss functions. Our hypothesis is that a representation that truly embodies the phonemic structure of speech will exhibit greater resilience, maintaining the spoken content even when faced with augmentations. Figure 1 visually depicts NAST.

We evaluate NAST’s invariance to signal variations (i.e.,

time-stretch, pitch-shift, additive-noise, and reverberation), encoding capabilities (ABX), together with zero-shot sequence modeling evaluations, i.e., sWUGGY, sBLIMP [20], and Spoken StoryCloze [21]. Results suggest NAST is comparable or superior to the evaluated baselines across all evaluation methods, considering various cluster numbers. We additionally analyze the learned representation considering speaker information and invariance to noise. These results shed light on the properties captured by the proposed speech tokenizer.

2. Method

The proposed method is comprised of three synergistic components: a frame-level Predictor E_l outputting logits \mathbf{p}_{E_l} across discrete units, a Residual Encoder E_g generating a global representation of the speech utterance, and a Decoder D reconstructing the original representation given the concatenated representations. A visual description of NAST can be seen in Figure 1.

The input is an internal representation obtained from the 9th layer of a pre-trained HuBERT model [1]. Formally, we define the representation of the speech utterance as $\mathbf{x} = (x_1, \dots, x_T)$ of T frames. Our goal is to establish a mapping $E_l(\mathbf{x}) = \mathbf{z} = [z_1, \dots, z_T]$, where each $z_i \in [1 \dots k]$ is a categorical variable representing one of k discrete units. Here, E_l denotes the quantizer model we seek to learn. In parallel, we aim to learn a global representation $E_g(\mathbf{x}) = \mathbf{u}$, capturing global information about the input sequence.

2.1. Network architecture

Predictor network. The predictor network, E_l , takes \mathbf{x} as input and predicts a discrete distribution over the learned units at each time step, denoted as $\mathbf{p}_{E_l}(\cdot | \mathbf{x}, t)$. That is, each frame $t \in T$ is associated with a k -dimensional logits vector in \mathbb{R}^k . By employing a Gumbel-SoftMax operation on the logits, we derive a one-hot encoded vector $\mathbf{1}_t$ in a differential manner, which we consider as a *local representation*. In essence, the one-hot vector $\mathbf{1}_t$ not only assigns the t -th frame to a specific unit but also contributes to the gradual shaping of each unit’s meaning. Through exposure to diverse speech patterns and contexts, the predictor learns to refine the mapping such that each unit increasingly reflects consistent aspects of spoken content.

In order to sample from the predicted logits, we leverage the Gumbel-SoftMax reparametrization trick [22], enabling the model to simulate discrete choice within a differentiable framework. Formally, the Gumbel-SoftMax distribution offers a continuous proxy for the categorical distribution. This continuity is achieved by introducing Gumbel noise to the logits associated with each category, followed by applying the SoftMax function:

$$\mathbf{1}_t(i) = \frac{\exp\left(\frac{\log(\pi_i) + g_i}{\tau}\right)}{\sum_{j=1}^k \exp\left(\frac{\log(\pi_j) + g_j}{\tau}\right)}, \quad \pi_i \triangleq \mathbf{p}_{E_l}(i | \mathbf{x}, t), \quad (1)$$

where π_i denotes the logit of the i -th unit, g_i is a Gumbel noise sample, k is the number of learned units, and τ is the temperature parameter that modulates the distribution’s discreteness. A low τ yields a more discrete-like distribution, mimicking a one-hot encoded vector, thus facilitating the transition from continuous logits to a quasi-one-hot representation.

Residual encoder. In opposite to the predictor module, which operates at the frame level, the residual encoder E_g aims to capture the global information within the signal. For that, we average the representations obtained by E_g over time. Formally, for a speech utterance \mathbf{x} , its *global representation* is given by: $\mathbf{u} \triangleq E_g(\mathbf{x}) = \frac{1}{T} \sum_{t=1}^T E_g(x_t)$.

This disentanglement facilitates the isolation of global attributes, thereby enabling the predictor to focus on modeling the local, content-driven nuances of speech.

2.2. Objective functions

The total objective function is a composite loss consisting of three key components: (i) Reconstruction loss, which aims to reconstruct the original HuBERT representation; (ii) Robustness loss, which ensures consistency of the model’s tokenization in the presence of augmentations that do not alter the core spoken content; and (iii) Diversity loss, designed to promote the use of a wide range of units within the model.

Reconstruction Loss. For each frame $t \in T$, the decoder D reconstructs the original HuBERT embedding, given as input the concatenated local and global representations, denoted as $\mathbf{1}_t$ and \mathbf{u} , respectively. $\mathcal{L}_{\text{recon}}$ is defined as:

$$\mathcal{L}_{\text{recon}}(E_l, E_g, D; \mathbf{x}) = \frac{1}{T} \sum_{t \in T} L_1(D(\mathbf{1}_t \oplus \mathbf{u}), x_t) \quad (2)$$

Robustness Loss. We wish to maintain consistent tokenization of a clean speech utterance \mathbf{x} across various augmentations that preserve the spoken content. In line with [19], we adopt an alignment-based strategy to enhance the invariance of NAST, utilizing a wide range of augmentations.

Formally, consider a clean speech utterance \mathbf{x} of T frames and its augmented version $\tilde{\mathbf{x}}$ of T' frames. Initially, both are fed into the predictor E_l , which outputs T and T' logit vectors for each, respectively. From the logit vectors of the clean signal, we derive T one-hot vectors, each representing the model’s chosen unit for the corresponding frame, forming the “ground truth” in this context. To address the temporal discrepancies induced by augmentations, we use linear interpolation to adjust the number of logit vectors from the augmented signal $\tilde{\mathbf{x}}$ to align with the number of one-hot vectors from the clean signal \mathbf{x} :

$$\mathbf{p}'_{E_l}(\cdot | \tilde{\mathbf{x}}) = \text{Interpolate}(\mathbf{p}_{E_l}(\cdot | \mathbf{x}), T) \quad (3)$$

Next, we make a frame-wise comparison between the aligned augmented logits, $\mathbf{p}'_{E_l}(\cdot | \tilde{\mathbf{x}}, 1) \dots \mathbf{p}'_{E_l}(\cdot | \tilde{\mathbf{x}}, T)$, and the clean signal’s one-hot vectors, $\mathbf{1}_1, \dots, \mathbf{1}_T$, as follows:

$$\mathbf{v}_t(i) \triangleq \frac{\exp(\mathbf{p}'_{E_l}(i | \tilde{\mathbf{x}}, t))}{\sum_{j=1}^k \exp(\mathbf{p}'_{E_l}(j | \tilde{\mathbf{x}}, t))} \quad (4)$$

$$\mathcal{L}_{\text{robust}} = \sum_{t=1}^T - \overbrace{\sum_{i=1}^k \mathbf{1}_t(i) \cdot \log(\mathbf{v}_t(i))}^{\text{CrossEntropy}(\mathbf{v}_t, \mathbf{1}_t)} \quad (5)$$

This comparison iteratively refines the unit selection, encouraging the model to yield similar distributions for both augmented and original signals, thus enhancing augmentation invariance.

Diversity Loss. To address a significant training challenge, where the model tends to “saturate” by favoring a limited set of units, we incorporate a Diversity Loss, drawing inspiration from the work of [3]. Applied to the clean speech utterance \mathbf{x} , we define the diversity loss as follows:

$$\mathcal{L}_{\text{diversity}} = \frac{1}{k} \sum_{i=1}^k \bar{p}_i \log(\bar{p}_i), \quad \bar{p}_i = \frac{1}{T} \sum_{t=1}^T \mathbf{1}_t(i) \quad (6)$$

Here, \bar{p}_i quantifies the average selection rate of the i -th unit by the predictor over the T frames within the clean utterance \mathbf{x} .

Overall, the final objective function is computed as a weighted sum of the three terms:

$$\mathcal{L}_{total} = \mathcal{L}_{recon} + \lambda_1 \cdot \mathcal{L}_{diversity} + \lambda_2 \cdot \mathcal{L}_{robust} \quad (7)$$

where λ_1 and λ_2 are tunable hyperparameters.

3. Experimental Setup

In all setups we consider the base version of HuBERT [1], using the 9th’ layer, operating at 50Hz, using 50, 100, and 200 clusters. We optimize, compare, and analyze the performance of various SpeechLMs (ULMs) learned over the discrete units, considering different setups. The training of both the ULMs and the quantizers is efficiently executed on a single NVIDIA A6000 GPU. We followed a similar setup as in [19], where we subject the speech signals to a suite of augmentations each chosen for its potential to simulate real-world signal variability and thereby refine the learned units’ robustness. Specifically, we consider: (i) **time-stretch** using a Phase Vocoder method [23] to stretch or shrink the time domain signal in the range [0.8, 1.2] without changing the pitch; (ii) **pitch-shifting** the speech signal by four semitones using the resampling method over the time-stretched signal [23]; (iii) **reverberation** following similar setting as [24] simulated via the pyroomacoustics [25] audio room simulations package; and (iv) **noise injection** using a randomly sampled Signal-to-Noise Ratio (SNR) in the range of [5, 15]. Background noises are sampled from the Deep Noise Suppression (DNS) challenge [26] which includes a diverse set of noise types from AudioSet [27], Freesound, [28], and Demand [29].

3.1. Model & Hyperparameters

Speech encoding. We use a learning rate of $1e - 4$ and a batch size of 16, employing the Adam optimizer [30]. The architecture of our network features Conformer blocks [31], each followed by a projection layer. The projection layer sizes for the decoder D and the residual encoder E_g are set to 768 and 256, respectively. The size of the predictor E_l projection depends on the number of units, k . Empirically determining the optimal configuration for each unit count involves assessing the number of layers, attention heads, kernel size, and feed-forward network (FFN) dimension. All quantizers are trained using the LS 960h dataset. We employed a weighted loss, where λ_1 and λ_2 denote the weights assigned to the diversity and robustness losses, respectively. We initiated the hyper-parameters to $\lambda_1 = 1$ and $\lambda_2 = 0.005$. We observed a sensitive interaction between the losses, where an increase in λ_2 tended to reduce the unit diversity. To address this, we carefully tuned both λ_1 and λ_2 upwards, ensuring that any adjustments were made in a controlled manner to prevent one loss from overshadowing the other.

unit-LM: For each number of clusters k , we train two types of uLM: one operating over our units and the other over units obtained via k-means quantization. The k-means quantizers are derived from the textless-lib [32]. These uLMs are based on the `transformer_lm_big` architecture implemented in `fairseq` [33], where each sample within the batch contains up to 4,096 units. During training, these models operate as causal language models on sequences of deduped units as in [5]. All language models are trained on a “clean” 6k-hour sub-sample of LibriLight [34].

4. Results

We evaluate NAST across two axes: (i) speech encoding evaluation; and (ii) sequence modeling evaluation. We additionally

Table 1: *Augmentation invariance results. UED is reported for NAST, k-means, and Gat et al., considering additive noise, time-stretch, reverberation, and pitch-shift. Results are reported for 50, 100, and 200 units.*

Units	Method	Augmentation UED ↓			
		Noise	Time-Stretch	Reverb	Pitch Shift
50	K-Means	29.74	39.61	28.25	44.33
	Gat, et al [19]	24.67	26.89	19.89	30.22
	NAST	9.51	17.26	9.82	16.47
100	K-Means	31.38	41.97	30.42	48.68
	Gat, et al [19]	25.06	29.72	21.31	32.84
	NAST	10.82	17.45	10.35	18.74
200	K-Means	33.34	45.59	32.89	53.14
	Gat, et al [19]	26.76	32.99	22.94	36.45
	NAST	11.88	21.36	13.86	22.97

provide an analysis to better highlight the properties of the proposed quantizer. We compare NAST to the commonly used k-means method [5, 7] and the method proposed by [19], considering 50, 100, and 200 clusters.

4.1. Speech encoding evaluation

We evaluate speech encoding considering both its phonetic discriminative capabilities using the ABX metric together with invariance to signal variations.

Augmentation invariance: We utilized the Unit Edit Distance UED metric proposed by [19] to evaluate the robustness of NAST to signal variations. This metric, based on the Levinstein Distance [35], measures the dissimilarity between the clean and augmented signals in terms of de-duplicated units. Ideally, a perfect spoken language quantizer would obtain a zero distance after deduplication. Table 1 presents the UED scores across various augmentations and k values. Notably, our method consistently outperforms previous works by a significant margin, indicating its superiority in handling these challenges.

ABX: The ABX task examines the discriminative phonetic abilities of the representation. It involves a pair of words differing by a single phoneme and a reference test word sharing a phoneme with one of the pair. It assesses whether the test phoneme is closer in representation to the correct or incorrect phoneme, expecting a shorter distance to the correct one. The ABX task is conducted in two setups: ‘within’ and ‘across’. ‘Within’ is evaluated on input data from the same speaker, while ‘across’ is evaluated on input data from different speakers. Table 2 shows the ABX results for both. We observed that the ABX-clean score was consistently on par with or exceeded the benchmarks established by other methods. Importantly, our approach demonstrated significant and steady enhancements in the more challenging ‘other’ split, which is distinguished by recordings that feature background noise and a variety of accents.

4.2. Sequence modeling evaluation

Under the modeling evaluation, we consider both the zero-resource speech metrics together with *Spoken Story Cloze*. To highlight the invariance of NAST to signal variations, we report results for two data versions: clean signals and noisy signals. The clean version is the original benchmark, while the noisy is an augmented version of the speech utterance.

Zero resource speech: We start by evaluating NAST using the zero resource speech metrics [20], i.e., sWUGGY, and sBLIMP. The sWUGGY metric requires detecting the real word from a pair of short utterances such as ‘brick’ vs. ‘blick.’ Similarly,

Table 2: *Modeling and encoding evaluation. We report results for NAST, k-means, and Gat et al. [19]. Results of [19] were taken from the paper as no code / pre-trained models were publicly available.*

Units	Method	sWUGGY \uparrow		sBLIMP \uparrow		TSC \uparrow		ABX (clean) \downarrow		ABX (other) \downarrow	
		clean	aug	clean	aug	clean	aug	within	across	within	across
50	K-Means	67.48	62.61	52.42	51.81	66.27	59.95	7.52	8.90	9.84	13.50
	Gat, et al [19]	67.59	-	53.68	-	-	-	6.63	7.55	9.53	12.14
	NAST	67.14	63.69	54.34	52.78	64.51	59.32	5.85	6.74	7.77	10.25
100	K-Means	67.75	64.46	51.96	51.82	67.18	60.55	6.37	7.72	8.4	12.29
	Gat, et al [19]	68.20	-	53.12	-	-	-	5.39	6.22	7.46	10.20
	NAST	73.35	70.01	55.86	54.72	64.13	61.78	5.20	5.90	6.92	8.73
200	K-Means	71.88	66.56	52.43	51.15	67.55	58.04	5.99	7.14	8.23	11.51
	Gat, et al [19]	70.68	-	54.91	-	-	-	5.19	6.00	7.18	9.70
	NAST	76.42	71.79	55.62	55.07	66.70	64.45	5.47	6.22	7.18	9.43

sBLIMP requires detecting the syntactically correct sentence from a pair of sentences. In both metrics, detection is done by comparing the probabilities of both sequences. As summarized in Table 2, NAST exhibited significant sWUGGY and sBLIMP improvements across most setups. As expected, these improvements are kept both under the clean and augmented versions.

Spoken StoryCloze: Next, we adapted the topic version of the Spoken StoryCloze Benchmark (TSC) [21] to assess our Unit Language Models’ ability to understand narrative continuity and capture intricate nuances. This benchmark involves distinguishing the correct ending from an adversarial one in a set of 4,000 five-sentence commonsense stories. Results are summarized in Table 2. When considering the clean version, NAST provides inferior performance to the k-means alternative, however, when considering the noisy version NAST consistently outperforms the k-means method.

4.3. Analysis

Noise invariance. To analyze the models’ sensitivity to augmentations, we progressively increase the level of augmentations and measure the impact on model performance. We measure the TSC performance as a function of noise levels, measured in SNR. As shown in Figure 2a, although the k-means alternative performs better under clean samples, as we increase the noise levels its performance rapidly decreases, this is in contrast to NAST which provides much more stable results.

Speaker probing. Lastly, we analyze the differences between global and local representations in capturing global information, specifically, we consider speaker information. For that, we perform speaker probing [32, 36]. This task assesses the model’s ability to identify an anonymized speaker ID based on their utterances, thus reflecting the extent to which speaker-specific information is retained in the representations. We compared NAST’s local representations against k-means clustering, in Figure 2b. Results suggest the local representation obtained by NAST contains less speaker information compared to the k-means one across all setups, whereas as we increase the number of clusters the gap increases. In contrast, when investigating the **global representations**, NAST achieves high accuracies, with **92.73%**, **98.24%**, and **97.67%** for **50**, **100**, and **200** units, respectively. These results emphasize the residual encoder’s efficacy in differentiating between local and global information.

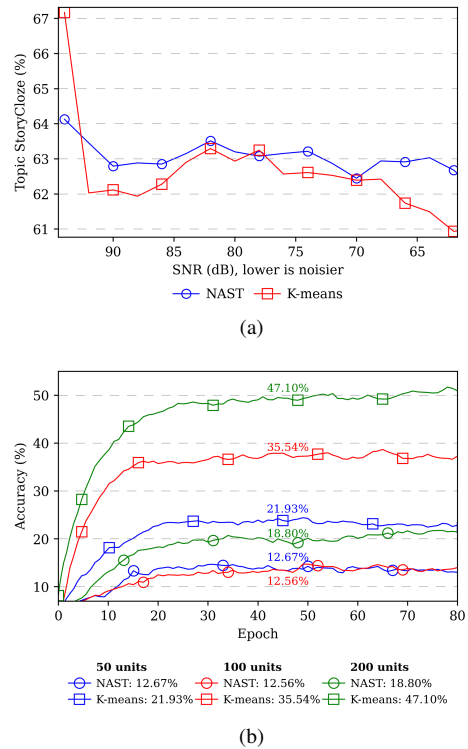


Figure 2: (a) TSC performance as a function of noise levels. Results are reported for both NAST and k-means using 100 clusters. (b) Speaker Probing for Local Representation: Classifiers trained for 100-epoch on LibriSpeech ‘dev-clean’.

5. Conclusion

This study introduces a novel speech tokenization method that enhances the robustness and performance of GSLM models. By adopting a dynamic strategy to map and refine speech units, our method departs from the traditional k-means clustering, introducing an innovative way to tokenize spoken language. Supported by comprehensive evaluations, our model demonstrates superior performance in handling diverse acoustic variations. Looking ahead, we intend to develop advanced and hierarchical tokenization techniques, enhancing our understanding and modeling of spoken language models.

Acknowledgements This research work was supported by ISF grant 2049/22.

6. References

- [1] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhota, 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*, 2021.
- [2] 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 Journal of Selected Topics in Signal Processing*, 2022.
- [3] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," 2020.
- [4] A. Mohamed, H.-y. Lee, L. Borgholt, J. D. Havtorn, J. Edin, C. Igel, K. Kirchhoff, S.-W. Li, K. Livescu, L. Maaløe *et al.*, "Self-supervised speech representation learning: A review," *IEEE Journal of Selected Topics in Signal Processing*, 2022.
- [5] K. Lakhota, E. Kharitonov, W.-N. Hsu, Y. Adi, A. Polyak, B. Bolte, T.-A. Nguyen, J. Copet, A. Baevski, A. Mohamed, and E. Dupoux, "On Generative Spoken Language Modeling from Raw Audio," *TACL*, 2021.
- [6] E. Kharitonov *et al.*, "Text-free prosody-aware generative spoken language modeling," *arXiv preprint arXiv:2109.03264*, 2021.
- [7] Z. Borsos, R. Marinier, D. Vincent, E. Kharitonov, O. Pietquin, M. Sharifi, O. Teboul, D. Grangier, M. Tagliasacchi, and N. Zeghidour, "Audiom: a language modeling approach to audio generation," *arXiv preprint arXiv:2209.03143*, 2022.
- [8] E. Kharitonov, D. Vincent, Z. Borsos, R. Marinier, S. Girgin, O. Pietquin, M. Sharifi, M. Tagliasacchi, and N. Zeghidour, "Speak, read and prompt: High-fidelity text-to-speech with minimal supervision," *arXiv preprint arXiv:2302.03540*, 2023.
- [9] A. Polyak, Y. Adi, J. Copet, E. Kharitonov, K. Lakhota, W.-N. Hsu, A. Mohamed, and E. Dupoux, "Speech resynthesis from discrete disentangled self-supervised representations," *arXiv preprint arXiv:2104.00355*, 2021.
- [10] Z. Wang, X. Zhu, Z. Zhang, Y. Lv, N. Jiang, G. Zhao, and L. Xie, "Selm: Speech enhancement using discrete tokens and language models," *arXiv preprint arXiv:2312.09747*, 2023.
- [11] H. Erdogan, S. Wisdom, X. Chang, Z. Borsos, M. Tagliasacchi, N. Zeghidour, and J. R. Hershey, "Tokensplit: Using discrete speech representations for direct, refined, and transcript-conditioned speech separation and recognition," *arXiv preprint arXiv:2308.10415*, 2023.
- [12] F. Kreuk *et al.*, "Textless speech emotion conversion using decomposed and discrete representations," *arXiv preprint arXiv:2111.07402*, 2021.
- [13] G. Maimon and Y. Adi, "Speaking style conversion in the waveform domain using discrete self-supervised units," in *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.
- [14] T. A. Nguyen, E. Kharitonov, J. Copet, Y. Adi, W.-N. Hsu, A. Elkahky, P. Tomasello, R. Algayres, B. Sagot, A. Mohamed *et al.*, "Generative spoken dialogue language modeling," *arXiv preprint arXiv:2203.16502*, 2022.
- [15] A. Lee, P.-J. Chen, C. Wang, J. Gu, X. Ma, A. Polyak, Y. Adi, Q. He, Y. Tang, J. Pino *et al.*, "Direct speech-to-speech translation with discrete units," *arXiv preprint arXiv:2107.05604*, 2021.
- [16] S. Popuri, P.-J. Chen, C. Wang, J. Pino, Y. Adi, J. Gu, W.-N. Hsu, and A. Lee, "Enhanced direct speech-to-speech translation using self-supervised pre-training and data augmentation," *arXiv preprint arXiv:2204.02967*, 2022.
- [17] A. Lee *et al.*, "Textless speech-to-speech translation on real data," in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2022, pp. 860–872.
- [18] Y. Wang, J. Bai, R. Huang, R. Li, Z. Hong, and Z. Zhao, "Speech-to-speech translation with discrete-unit-based style transfer," *arXiv preprint arXiv:2309.07566*, 2023.
- [19] I. Gat, F. Kreuk, T. A. Nguyen, A. Lee, J. Copet, G. Synnaeve, E. Dupoux, and Y. Adi, "Augmentation invariant discrete representation for generative spoken language modeling," in *Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023)*. Association for Computational Linguistics, 2023, pp. 465–477.
- [20] T. A. Nguyen, M. de Seyssel, P. Rozé, M. Rivière, E. Kharitonov, A. Baevski, E. Dunbar, and E. Dupoux, "The zero resource speech benchmark 2021: Metrics and baselines for unsupervised spoken language modeling," in *NeurIPS – Self-Supervised Learning for Speech and Audio Processing Workshop*, 2020.
- [21] M. Hassid, T. Remez, T. A. Nguyen, I. Gat, A. Conneau, F. Kreuk, J. Copet, A. Defossez, G. Synnaeve, E. Dupoux *et al.*, "Textually pretrained speech language models," *arXiv preprint arXiv:2305.13009*, 2023.
- [22] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," *arXiv preprint arXiv:1611.01144*, 2016.
- [23] T. Karrer, E. Lee, and J. O. Borchers, "Phavorit: A phase vocoder for real-time interactive time-stretching," in *ICMC*, 2006.
- [24] S. E. Chazan, L. Wolf, E. Nachmani, and Y. Adi, "Single channel voice separation for unknown number of speakers under reverberant and noisy settings," in *ICASSP*, 2021.
- [25] R. Scheibler, E. Bezzam, and I. Dokmanić, "Pyroomacoustics: A python package for audio room simulation and array processing algorithms," in *ICASSP*, 2018.
- [26] C. K. A. Reddy *et al.*, "The interspeech 2020 deep noise suppression challenge: Datasets, subjective testing framework, and challenge results," 2020.
- [27] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *ICASSP*, 2017.
- [28] F. Font Corbera, G. Roma Trepat, and X. Serra, "Freesound technical demo," in *MM'13. Proceedings of the 21st ACM international conference on Multimedia; 2013 Oct 21-25; Barcelona, Spain. New York: ACM; 2013. p. 411-2*. ACM Association for Computer Machinery, 2013.
- [29] J. Thiemann, N. Ito, and E. Vincent, "Demand: a collection of multi-channel recordings of acoustic noise in diverse environments," in *Proc. Meetings Acoust.*, 2013.
- [30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [31] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented transformer for speech recognition," 2020.
- [32] E. Kharitonov, J. Copet, K. Lakhota, T. A. Nguyen, P. Tomasello, A. Lee, A. Elkahky, W.-N. Hsu, A. Mohamed, E. Dupoux *et al.*, "textless-lib: a library for textless spoken language processing," *arXiv preprint arXiv:2202.07359*, 2022.
- [33] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli, "fairseq: A fast, extensible toolkit for sequence modeling," 2019.
- [34] J. Kahn, M. others Rivière, W. Zheng, E. Kharitonov, Q. Xu, P.-E. Mazaré, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen *et al.*, "Libri-light: A benchmark for asr with limited or no supervision," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7669–7673.
- [35] L. Yujuan and L. Bo, "A normalized levenshtein distance metric," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 6, pp. 1091–1095, 2007.
- [36] Y. Adi, N. Zeghidour, R. Collobert, N. Usunier, V. Liptchinsky, and G. Synnaeve, "To reverse the gradient or not: An empirical comparison of adversarial and multi-task learning in speech recognition," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 3742–3746.