INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

TokenSplit: Using Discrete Speech Representations for Direct, Refined, and Transcript-Conditioned Speech Separation and Recognition

Hakan Erdogan[1], *Scott Wisdom*[1], *Xuankai Chang*[2*], *Zalan Borsos*[1], *Marco Tagliasacchi*[1], *Neil Zeghidour*[1], *John R. Hershey*[1]

[1]Google Research
[2]Carnegie Mellon University

{hakanerdogan, scottwisdom, zborsos, mtagliasacchi, neilz, johnhershey}@google.com, xuankaic@andrew.cmu.edu

Abstract

We present TokenSplit, a speech separation model that acts on discrete token sequences. The model is trained on multiple tasks simultaneously: separate and transcribe each speech source, and generate speech from text. The model operates on transcripts and audio token sequences and achieves multiple tasks through masking of inputs. The model is a sequence-to-sequence encoder-decoder model that uses the Transformer architecture. We also present a "refinement" version of the model that predicts enhanced audio tokens from the audio tokens of speech separated by a conventional separation model. Using both objective metrics and subjective MUSHRA listening tests, we show that our model achieves excellent performance in terms of separation, both with or without transcript conditioning. We also measure the automatic speech recognition (ASR) performance and provide audio samples of speech synthesis to demonstrate the additional utility of our model.

Index Terms: speech separation, speech recognition, speech synthesis, discrete representations, language modeling

1. Introduction

Sound separation is essential for audio scene understanding and manipulation. Speech separation is a sub-task of sound separation which has its own importance. It aims to isolate different overlapping speech sources in a mixture, such as in a recording of a meeting. Many methods have been proposed recently that typically involve training a neural network on artificial mixtures of isolated speech to estimate individual sources [1, 2, 3]. To address the lack of labeled data in real environments, unsupervised training methods [4, 5, 6] have been proposed. Significant progress has been achieved recently; however, it is well-known that single-channel speech separation models can introduce unwanted distortions that impact perceptual quality and may be harmful for downstream tasks like speech recognition [7, 8, 9].

In this work, we propose a novel speech separation model that uses discrete token sequences and has additional capabilities. Our method uses three kinds of discrete tokens: (1) SoundStream [10] tokens as acoustic tokens, (2) w2v-BERT [11] tokens as semantic tokens, (3) transcripts represented as character sequences. Our model is based on a sequence-to-sequence encoder-decoder architecture. The semantic and acoustic tokens of the speech mixture as well as transcripts of each source in the mixture are provided at the input (see left panel of Figure 1). As the target, the model tries to predict transcripts, semantic and acoustic tokens of each source. During training, one or more kinds of input tokens are probabilistically replaced with mask tokens to simulate different modes of operation. After training, the model can be used to do speech separation and recognition given a speech mixture. The model can also do separation conditioned on one or both of the source transcripts, and speech synthesis when only transcripts are provided. Separation performance of the model is additionally improved using "refinement", where the output of a conventional audio separation model is refined by token-based autoregressive generation.

To evaluate our model, experiments are carried out on the Libri2Mix dataset [12]. The results show that our model can achieve excellent performance in terms of separation quality supported by subjective listening tests, both with and without transcript conditioning. As a by-product, TokenSplit also produces the ASR transcripts corresponding to both speakers. In our evaluation we observe that the quality of the predicted transcripts is not at the same level as a dedicated ASR model operating on the separated audio. In this paper we focus on separation performance, but we plan to improve ASR quality in future work.

2. Relation to Prior Work

Generative approaches for speech enhancement and speech separation [13, 14, 15] consist of an encoder and separator that predicts intermediate representations of individual signals, from which separated signals can be generated by a vocoder. Notably, [15] uses a discrete token representation. Such approaches can generate separated speech without interference. However, the quality of generative approaches is hampered by deviation from the phonetic content, speaker characteristics, or speaking style of the original speaker.

In [10], an end-to-end neural audio codec model was proposed, called SoundStream. The model is similar to a VQ-VAE [16, 17] and VQ-GAN [18]. The SoundStream codec has been used in a recent framework of audio generation, called AudioLM [19]. Semantic tokens are obtained from a w2v-BERT model [11], and acoustic tokens from a SoundStream [10] encoder. SoundStream provides a deterministic codec that maps audio to discrete tokens and vice versa, avoiding the training and use of a separate vocoder unlike previous generative separation/enhancement approaches. AudioLM was originally applied to the task of continuation of speech and piano audio, where the model generates audio that is consistent with an initial short prompt. Recently, this framework has also been applied to generating music from text descriptions [20], accompaniment for singing [21], and text to speech (TTS) [22].

We use models introduced in previous work such as SoundStream [10] and w2v-BERT [11] to extract tokens. We also use ideas from AudioLM [19] and apply them to the separation problem. Our models can perform a variety of tasks, including unconditional speech separation, transcript-conditioned speech

[*] Work done during an internship at Google.

10.21437/Interspeech.2023-2069

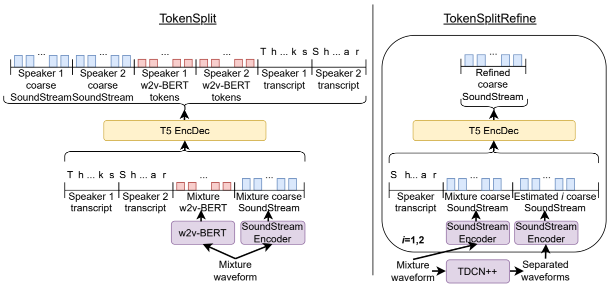

Figure 1: *Illustration of models. Left: TokenSplit separation/recognition/synthesis, with optional transcript conditioning. Right: TokenSplitRefine method, where separated sources from a conventional separation model are enhanced with optional transcript conditioning. Note that TokenSplitRefine is run for each source independently, indicated by the plate notation.*

separation, ASR of overlapping speakers and multi-speaker TTS. Nevertheless, our focus is separation and transcript-conditioned separation capabilities of the model.

3. Method

In this section, we present the details of our model, as shown in Fig. 1. We denote a mixture containing overlapping speech from N speakers as $\mathbf{y} = \sum_{i=1}^{N} \mathbf{x}_i$, where $\mathbf{y} \in \mathbb{R}^T$ is the mixture of length T samples and $\mathbf{x}_i \in \mathbb{R}^T$ is the $i-$th speaker's speech. In this paper, we consider the 2-speaker case, $N = 2$.

3.1. Token Extraction

Motivated by the successes of AudioLM [19], we explore using both semantic and acoustic tokens in our model. Since semantic tokens carry phonetic information and acoustic tokens represent the audio signal, we expect them to complement each other.

Semantic tokens are computed using w2v-BERT [11], which is a self-supervised learning model trained with both a masked language modeling (MLM) loss and a contrastive loss. We use a pre-trained model to map the input mixture audio waveform to embeddings extracted from an intermediate layer, which contain rich linguistic information. To discretize these embeddings, we train a *k*-means model on the embeddings with K clusters. We extract the semantic tokens as follows:

$$\mathbf{S}_{\text{mix}} = \text{Discretize}\left(\text{w2v-BERT}(\mathbf{y})\right), \quad (1)$$
$$\mathbf{S}_i = \text{Discretize}\left(\text{w2v-BERT}(\mathbf{x}_i)\right),\ i \in \{1, \ldots, N\}, \quad (2)$$

where $\mathbf{S}_{\text{mix}}$ and $\mathbf{S}_i$ are semantic token sequences for input mixture and clean source i, respectively, and T_S is the sequence length. The “Discretize” operation maps each embedding to the index of the closest *k*-means codebook entry [19].

For acoustic token extraction, we use a SoundStream model [10], which is a state-of-the-art end-to-end neural audio codec at low bitrates. We denote the acoustic tokens as follows:

$$\mathbf{A}_{\text{mix}} = \text{SoundStream}\left(\mathbf{y}\right), \quad (3)$$
$$\mathbf{A}_i = \text{SoundStream}\left(\mathbf{x}_i\right),\ i \in \{1, \ldots, N\}, \quad (4)$$

where $\mathbf{A}_{\text{mix}}$ and are acoustic token sequences for input overlapping speech and clean sources, with length T_A. The acoustic token at each time step is a vector of Q dimensions.

We compute transcripts from clean signals using a pre-trained ASR model instead of using ground-truth transcripts, mainly because the source speech signals are clipped to a fixed length which may chop words in the boundaries and the ground truth transcripts for them are not immediately available. We obtain $\mathbf{W}_i = \text{ASR}\left(\mathbf{x}_i\right),\ i \in \{1, \ldots, N\}$, where $\mathbf{W}_i$ are encoded transcript tokens for clean sources i, with length L_i.

3.2. Encoder-Decoder Processing

We use a sequence-to-sequence encoder-decoder (EncDec) Transformer architecture for our model. Cross-entropy on tokens is used as the loss function. To avoid very long sequences in the input and output, as in AudioLM, we experimented with using two stages to first predict coarse, then fine, acoustic tokens. But in our experiments, we found that just a single stage predicting coarse acoustic tokens was sufficient. Thus, we only present results with a single stage, and report results with two stages in supplementary material [23].

The model takes the transcripts of each source, semantic tokens of overlapping speech, and coarse acoustic tokens of overlapping speech as input. The token values are made separate and unique before feeding into the network by shifting the token values to be in a certain range for each kind of token. We use a special separator token to separate different kinds of tokens. We apply random masking on each kind of input token sequence, so some of the sequences are fully replaced with special mask tokens. The flexibility to mask any combination of tokens allows us to train the model for a wide variety of tasks. Transcripts are padded to a fixed length of 150 with the same mask token when they are not fully replaced by mask tokens. We also apply a separate type of masking where we replace a contiguous subregion with mask tokens as well. The model learns the following mapping:

$$\left[\hat{\mathbf{A}}_{1:N}^{\leq Q''}, \hat{\mathbf{S}}_{1:N}, \hat{\mathbf{W}}_{1:N},\right] = \text{EncDec}[\text{Mask}(\mathbf{W}_1), \cdot\cdot, \text{Mask}(\mathbf{W}_N), \text{Mask}(\mathbf{S}_{\text{mix}}), \text{Mask}(\mathbf{A}_{mix}^{\leq Q'})], \quad (5)$$

where $\text{Mask}(\cdot)$ represents the masking function, $\mathbf{A}_{mix}^{\leq Q'}$ denotes the coarse acoustic tokens of noisy speech and $\hat{\mathbf{A}}_{1:N}^{\leq Q''}$ denotes the predicted coarse acoustic tokens of each source from 1 to N. Note that Q'' can be different than Q'.

To generate audio waveforms, the coarse acoustic tokens are transformed by the SoundStream decoder.

3.3. Model Capabilities

In this part, we discuss the capability of our proposed model about joint separation, recognition and synthesis of overlapping speech through discrete representations. Consider three possible use cases to define different "inference modes" of the model.

1. **Separation and recognition.** If the transcripts $\mathbf{W}_{1:N}$ are masked out at the input, the model can only see semantic tokens and acoustic tokens of overlapping speech. In such cases, the model performs speech separation and recognition.
2. **Transcript-conditioned separation and recognition.** One or more of the transcripts are fed as input as well as the input semantic and acoustic tokens. In this case, the model can leverage the text information to predict acoustic tokens of each source.
3. **Multi-speaker text-to-speech synthesis.** This is a side effect of this model where it can be used as a 2-speaker-TTS model when providing one or more of the input transcripts without any audio input and generating speech based on the transcript(s).

3.4. TokenSplitRefine

We also consider using a variant of our system, which we call *TokenSplitRefine*, to enhance the output of a traditional audio-only separation system, such as a TDCN++ [24]. The outputs of this initial separation model are fed to a downstream TokenSplitRefine model, which can use optional transcript conditioning. An illustration of this setup is shown in the right panel of Figure 1. We did not use semantic tokens for the refinement model since the benefit was minimal. We predict the tokens of an enhanced source from optional source transcript tokens, mixture acoustic tokens, and separated source tokens. TokenSplitRefine can clean up distortion and artifacts present in the separated estimates. Note that this model is similar to some target speaker/speech extraction methods where an utterance of a target speaker is used to extract that speaker from a mixture. However, here we use an initial extraction of that speaker's speech as an input instead. This method is also similar to iterative architectures [25] and generative post-processing [26] used for sound separation problems.

4. Experiments

4.1. Training

TokenSplit models are trained using 10 million 3s long synthetic fully-overlapped speech mixtures which we mixed from random sources in the Libri-Light dataset [27] with each gain sampled from $\mathcal{N}(-10, 4)$ dB. We used a relatively short fixed block of 3 seconds for training and testing our models since there is a limit on the token length imposed by the Transformer's limited memory. The 3s blocks were obtained from the beginning of the variable-length mixture. In addition, our models have multiple outputs, and we need to keep positional encodings consistent across multiple runs through the model which is easier to achieve with a fixed duration input and output.

To extract semantic tokens, we use w2v-BERT XL [11] with 0.6B parameters. Following AudioLM [19], we select the 7th layer in the MLM module of w2v-BERT XL to extract the embeddings and $K = 1024$ clusters for the *k*-means. These are trained on Libri-Light [27], 60k hours of read English speech.

The SoundStream model used in this work uses a similar architecture as in previous work [19], however, it downsamples the 16 kHz inputs to 25 Hz embeddings. The vector quantizer has 40 layers and each layer uses a codebook size of 64. These 40 layers representation is mapped to a 20 layer representation with a 4096 codebook size for processing with the TokenSplit model. The resulting bitrate is 6000bps. We split the input acoustic token at level $Q' = 10$, and we use levels up to $Q'' = 6$ for coarse target tokens. The SoundStream model is trained on a combination of 8300 hours of Libri-Light speech mixtures and about 900 hours of LibriVox clean speech, so that it can represent both clean speech and speech mixtures well.

To obtain the transcripts used both as input and target in the model, we used a multi-lingual ASR model which has a 16 layer Conformer encoder and a 4 layer RNN decoder which was trained to recognize speech in 22 languages. The transcripts are encoded with character encoding and padded with mask tokens to a fixed length of 150 characters, enough to cover 3s blocks.

We use T5X [28] to train our proposed models. The Transformer-based encoder-decoder architecture is used in all stages, and has 267M parameters. The encoder and decoder contain 12 layers each, with 12 attention heads, embedding dimension of 768, feed-forward layer dimension of 2048 and dropout of 0.3. We used learnable absolute positional embeddings. Learnable relative position biases were added to each attention matrix as well. While training, we applied masking patterns to inputs with pre-assigned probabilities, detailed in our supplementary material [23]. Model vocabulary included 4 special tokens, 256 transcript tokens, 1024 semantic tokens and $4096Q$ acoustic tokens. Training was done for 1 million steps on 16 Google Cloud TPU v4 chips with the Adafactor optimizer [29] with batch size 256 and learning rate 0.5 with 10000-step warmup and inverse square-root decay. TokenSplit trains in about 7 days, and TokenSplitRefine in about 5 days.

Table 1: *Results on 3s and variable-length Libri2Mix. Column "Transcript" indicate which sources' transcripts are provided at the inputs to the models.*

Model	Transcript	SI-SNRi	DNSMOS	ViSQOL	DWER/WER	DCER/CER
			Libri2Mix test (3s)			
Mixture	–	0.0	3.39	1.50	86.6	62.2
TDCN++	–	12.3	2.96	2.11	25.1	14.9
TokenSplit	No	-3.5	3.50	2.04	26.6	15.4
TokenSplit	Non-target	-3.6	3.51	2.05	26.6	15.3
TokenSplit	Target	-3.5	3.51	2.11	12.1	6.4
TokenSplit	Both	-3.5	3.51	2.12	12.2	6.5
TokenSplitRefine	No	0.3	3.51	2.29	17.2	9.5
TokenSplitRefine	Target	0.3	3.51	2.33	10.9	5.7
			Libri2Mix test (variable-length)			
Mixture	–	0.0	3.57	1.69	87.3 / 87.6	66.5 / 67.0
TDCN++	–	16.0	3.29	2.42	13.0 / 21.6	7.3 / 13.2
TokenSplit	No	-2.9	3.61	2.06	21.1 / 28.3	12.1 / 17.5
TokenSplitRefine	No	0.3	3.62	2.25	13.6 / 22.4	7.5 / 13.8

Typically, separation models are trained with a permutation invariant loss function [1, 2]. Here, we did not do that, since teacher-forced training is used during training and autoregressive sampling is used during inference. The model is clued about the target source to decode from the previous sampled tokens. During inference, we sample the first token with model provided probabilities and this sample most likely decides which source is decoded first.

4.2. Evaluation

We tested the models' performance on the test set of the Libri2Mix [12], sampled at 16 kHz. We present results on the first 3 seconds of the test set, as well as on the original variable-length test set using "partial overlap decoding" (section 4.4).

To measure signal quality, we use scale-invariant signal-to-noise ratio improvement (SI-SNRi) [30], Virtual Speech Quality Objective Listener (ViSQOL) [31], and the non-intrusive Deep Noise Suppression Mean Opinion Score (DNSMOS) [32]. We also use ASR-based metrics. Since we do not always have ground-truth transcripts (e.g. for 3s blocks), we use differential word error rate (DWER) [25] and differential character error rate (DCER), which use ASR on ground-truth audio as ground-truth transcript. For variable-length Libri2Mix, we also compute standard WER and CER using ground-truth transcripts. Note that since the TokenSplit(Refine) models use autoregressive sampling at their output, metrics such as SI-SNRi that rely on exact sample alignment between references and estimates are not suitable. Frame-based ViSQOL and lexical content-based DWER/WER are also not perfect, but have more tolerance for such generative sampling.

4.3. Libri2Mix 3s Results

For the first 3s of Libri2Mix test clips, we compare the separation performance of TokenSplit models to an audio-only TDCN++ model [24] in the top half of Table 1. This TDCN++ model was trained on the Libri2Mix training set, different from the Libri-Light-based training set we used for TokenSplit. The Libri-Light training set is larger but more mismatched to Libri2Mix, since it contains noisier speech.

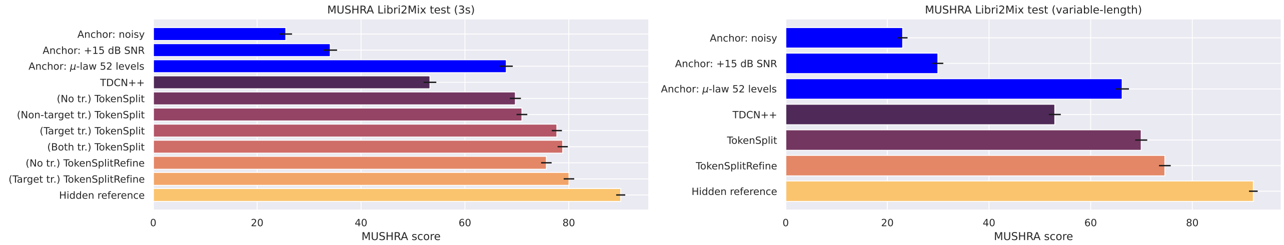

Figure 2: *Results of MUSHRA listening tests on first 3s of Libri2Mix test set (left) and original variable-length Libri2Mix test (right). For input transcript conditioning, "(No tr.)" is none, "(Non-target tr.)" is non-target transcript, "(Target tr.)" is target source transcript, and "(Both tr.)" is transcripts of both sources.*

We investigated the performance of different variants of our model that use different orders of outputs and/or missing modalities, such as some variants not using semantic tokens or transcripts. We list the variants we used and their performances in our supplementary material [23]. The best variant used the following order of output tokens: $\mathbf{A}_1 + \mathbf{A}_2 + \mathbf{S}_1 + \mathbf{S}_2 + \mathbf{W}_1 + \mathbf{W}_2$ and used all inputs of $\mathbf{W}_1 + \mathbf{W}_2 + \mathbf{S}_m + \mathbf{A}_m$. This variant performed the best when considering DNSMOS and DWER/DCER. The other variants generating text or semantic tokens first did not do as well in separation. The variant without semantic tokens achieved a DWER of 27.9% compared to 26.6% of the best variant for coarse audio, showing that semantic tokens may not be that important for this problem.

Table 1 also includes results for TokenSplitRefine described in section 3.4, with or without transcript conditioning. Notice that transcript conditioning improves performance significantly, as expected. TokenSplitRefine output is much better as compared to the TDCN++ output in terms of DNSMOS and DWER/DCER performances.

4.4. Libri2Mix Variable-Length Results

We performed variable-length inference through partial overlap decoding. We used 3s blocks with an overlap of 1s. The last 1s of the previous block's output is used to prompt the next block's decoding. This is possible in T5X by fixing the outputs at certain positions in the output sequence. Note that we prompted each output type (acoustic, semantic tokens) separately by the last 1s of the previous block result, so that the prompt is not only at the beginning part of the decoding. Transcripts were not prompted. Results are in the bottom half of Table 1.

Inspecting the results, we see that the performance of TokenSplit and TokenSplitRefine are better than the TDCN++ model in terms of DNSMOS, but worse in terms of DWER and DCER. This shows that there is room for improvement for the variable-length inference case. We think this difference can be due to mismatch with 3s training data. Another reason could be increased chance of sampling errors when generating variable-length data using partial overlap decoding.

4.5. Listening Tests

Since our objective metrics are imperfect (section 4.2), we performed MUSHRA listening tests [33] to measure the subjective quality of our models to provide a more reliable measurement of relative quality. 128 examples from the Libri2Mix test set were used, and there are 256 total examples rated since there are two sources per mixture. Each example was evaluated by 5 raters through an online interface and were required to wear headphones. We ran one listening test with variable-length audio (88 raters), and another listening test with the first 3 seconds of the same variable-length clips (97 raters).

As MUSHRA anchors, we used the input mixture, the input mixture with a 15 dB boost in SNR for the target source to simulate interference reduction, and μ-law encoding with 52 levels to simulate distortion of the source. We compare the conventional TDCN++ masking-based separation model against TokenSplit and TokenSplitRefine.

Results are shown in Figure 2. The TokenSplitRefine models are a clear winner, with their performance improving even more with transcript conditioning. All TokenSplit(Refine) models outperform the conventional TDCN++ model. Thus, our proposed model provides excellent subjective separation performance, confirming the quality measured by objective metrics. Audio demos of MUSHRA audio are provided online [23].

4.6. Recognition and TTS Performance of the TokenSplit Separation Model

For ASR with our model, we measured a DWER of 71.9% for the model's transcript predictions, which is more than double the DWER of 26.6% for the standalone ASR model applied to audio decoded from model's coarse SoundStream token predictions. We attribute this to conventional ASR models being trained with much large corpora that includes clean speech. In contrast, TokenSplit is only trained with much less training data that only contains mixtures. In a sense, signal separation is an easier problem than overlapping speech recognition. TokenSplit's ASR performance can likely be improved by increasing training data, using both clean and mixed speech for training, and optimizing transcript encodings.

For TTS, we did not evaluate the output of TokenSplit using objective or subjective metrics, but informal listening (see provided audio demos online [23]) indicates the model can generate reasonable outputs. We notice that TokenSplit randomly chooses two speakers to utter the input transcripts, and speaker identities and style remain constant throughout.

5. Conclusion

In this work, we introduced a multi-purpose model that can separate multiple sources from an input mixture audio with optional transcripts. Instead of using a time-domain or frequency-domain continuous-space inputs, we use discrete tokens as inputs and targets to train our model. In this novel model, we combine the semantic, acoustic and transcript tokens together as inputs and targets. A refinement version of the model takes pre-separated sources and refines them to be artifact free. Experiments on the Libri2Mix test set show that our models can achieve excellent performance compared to a conventional time-domain separation model, especially reducing artifacts and achieving high human-rated quality. Our model can also perform ASR and TTS, but more work is required to improve performance on these tasks.

6. References

[1] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *Proc. ICASSP*, 2016, pp. 31–35.

[2] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *Proc. ICASSP*, 2017, pp. 241–245.

[3] Y. Luo and N. Mesgarani, "Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 8, pp. 1256–1266, 2019.

[4] S. Wisdom, E. Tzinis, H. Erdogan, R. Weiss, K. Wilson, and J. Hershey, "Unsupervised sound separation using mixture invariant training," in *Advances in Neural Information Processing Systems*, 2020.

[5] S. Wisdom, A. Jansen, R. J. Weiss, H. Erdogan, and J. R. Hershey, "Sparse, efficient, and semantic mixture invariant training: Taming in-the-wild unsupervised sound separation," in *Proc. WASPAA*, 2021, pp. 51–55.

[6] E. Tzinis, Y. Adi, V. K. Ithapu, B. Xu, P. Smaragdis, and A. Kumar, "RemixIT: Continual self-training of speech enhancement models via bootstrapped remixing," *IEEE Journal of Selected Topics in Signal Processing*, 2022.

[7] T. Menne, R. Schlüter, and H. Ney, "Investigation into joint optimization of single channel speech enhancement and acoustic modeling for robust ASR," in *Proc. ICASSP*, 2019, pp. 6660–6664.

[8] S.-J. Chen, A. S. Subramanian, H. Xu, and S. Watanabe, "Building state-of-the-art distant speech recognition using the chime-4 challenge with a setup of speech enhancement baseline," in *Proc. Interspeech*, 2018, pp. 1571–1575.

[9] K. Iwamoto, T. Ochiai, M. Delcroix, R. Ikeshita, H. Sato, S. Araki, and S. Katagiri, "How bad are artifacts?: Analyzing the impact of speech enhancement errors on ASR," in *Proc. Interspeech*, 2022, pp. 5418–5422.

[10] 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, 2021.

[11] 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. IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2021, pp. 244–250.

[12] J. Cosentino, M. Pariente, S. Cornell, A. Deleforge, and E. Vincent, "LibriMix: An open-source dataset for generalizable speech separation," *arXiv preprint arXiv:2005.11262*, 2020.

[13] S. Maiti and M. I. Mandel, "Parametric resynthesis with neural vocoders," in *Proc. WASPAA*. IEEE, 2019, pp. 303–307.

[14] Q. Liu, P. J. Jackson, and W. Wang, "A speech synthesis approach for high quality speech separation and generation," *IEEE Signal Processing Letters*, vol. 26, no. 12, pp. 1872–1876, 2019.

[15] J. Shi, X. Chang, T. Hayashi, Y.-J. Lu, S. Watanabe, and B. Xu, "Discretization and re-synthesis: an alternative method to solve the cocktail party problem," *arXiv preprint arXiv:2112.09382*, 2021.

[16] A. Van Den Oord, O. Vinyals *et al.*, "Neural discrete representation learning," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[17] A. Razavi, A. Van den Oord, and O. Vinyals, "Generating diverse high-fidelity images with vq-vae-2," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[18] P. Esser, R. Rombach, and B. Ommer, "Taming transformers for high-resolution image synthesis," in *Proc. CVPR*, 2021, pp. 12 873–12 883.

[19] Z. Borsos, R. Marinier, D. Vincent, E. Kharitonov, O. Pietquin, M. Sharifi, O. Teboul, D. Grangier, M. Tagliasacchi, and N. Zeghidour, "AudioLM: a language modeling approach to audio generation," *arXiv preprint arXiv:2209.03143*, 2022.

[20] A. Agostinelli, T. I. Denk, Z. Borsos, J. Engel, M. Verzetti, A. Caillon, Q. Huang, A. Jansen, A. Roberts, M. Tagliasacchi *et al.*, "MusicLM: Generating music from text," *arXiv preprint arXiv:2301.11325*, 2023.

[21] C. Donahue, A. Caillon, A. Roberts, E. Manilow, P. Esling, A. Agostinelli, M. Verzetti, I. Simon, O. Pietquin, N. Zeghidour *et al.*, "SingSong: Generating musical accompaniments from singing," *arXiv preprint arXiv:2301.12662*, 2023.

[22] 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.

[23] *TokenSplit project website*, https://google-research.github.io/sound-separation/papers/tokensplit.

[24] I. Kavalerov, S. Wisdom, H. Erdogan, B. Patton, K. W. Wilson, J. L. Roux, and J. R. Hershey, "Universal sound separation," in *Proc. WASPAA*, 2019, pp. 175–179.

[25] Z.-Q. Wang, H. Erdogan, S. Wisdom, K. Wilson, D. Raj, S. Watanabe, Z. Chen, and J. R. Hershey, "Sequential multi-frame neural beamforming for speech separation and enhancement," in *Proc. SLT*, 2021, pp. 905–911.

[26] N. Schaffer, B. Cogan, E. Manilow, M. Morrison, P. Seetharaman, and B. Pardo, "Music separation enhancement with generative modeling," in *Proc. ISMIR*, 2022.

[27] J. Kahn, M. 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 *Proc. ICASSP*, 2020, pp. 7669–7673.

[28] A. Roberts, H. W. Chung, A. Levskaya, G. Mishra, J. Bradbury, D. Andor, S. Narang, B. Lester, C. Gaffney, A. Mohiuddin, C. Hawthorne, A. Lewkowycz, A. Salcianu, M. van Zee, J. Austin, S. Goodman, L. B. Soares, H. Hu, S. Tsvyashchenko, A. Chowdhery, J. Bastings, J. Bulian, X. Garcia, J. Ni, A. Chen, K. Kenealy, J. H. Clark, S. Lee, D. Garrette, J. Lee-Thorp, C. Raffel, N. Shazeer, M. Ritter, M. Bosma, A. Passos, J. Maitin-Shepard, N. Fiedel, M. Omernick, B. Saeta, R. Sepassi, A. Spiridonov, J. Newlan, and A. Gesmundo, "Scaling up models and data with `t5x` and `seqio`," *arXiv preprint arXiv:2203.17189*, 2022.

[29] N. Shazeer and M. Stern, "Adafactor: Adaptive learning rates with sublinear memory cost," in *Proc. ICML*, 2018, pp. 4596–4604.

[30] J. L. Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "SDR - half-baked or well done?" in *Proc. ICASSP*, 2019, pp. 626–630.

[31] 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. QoMEX*, 2020, pp. 1–6.

[32] C. K. A. Reddy, V. Gopal, and R. Cutler, "DNSMOS: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors," in *Proc. ICASSP*, October 2020, pp. 6493–6497.

[33] ITU, "Method for the subjective assessment of intermediate quality level of audio systems," *BS.1534*, 2014.