TSELM: Target Speaker Extraction using Discrete Tokens and Language Models

Beilong Tang[1][0009−0000−7941−8159], Bang Zeng[1][0009−0007−6934−0950], and Ming Li[1][0000−0002−6406−1983]

Duke Kunshan University
{beilong.tang,zeng.bang,ming.li369}@dukekunshan.edu.cn

Abstract. We propose TSELM, a novel target speaker extraction network that leverages discrete tokens and language models. TSELM utilizes multiple discretized layers from WavLM as input tokens and incorporates cross-attention mechanisms to integrate target speaker information. Language models are employed to capture the sequence dependencies, while a scalable HiFi-GAN is used to reconstruct the audio from the tokens. By applying a cross-entropy loss, TSELM models the probability distribution of output tokens, thus converting the complex regression problem of audio generation into a classification task. Experimental results show that TSELM achieves excellent results in speech quality and comparable results in speech intelligibility.

Keywords: target speaker extraction · audio discretization · language models · speech separation · self-supervised models

1   Introduction

In contrast to blind speech separation, which seeks to isolate individual utterances from a mixture of known speakers [20], Target Speaker Extraction (TSE) focuses on extracting only the voice of the target speaker in a mixture using auxiliary information [40]. Current models are predominantly discriminative, employing masking strategies to directly minimize the distance between the estimated and clean speech signals [11, 16, 27, 39]. However, these discriminative approaches often struggle to generalize to unseen data and may introduce undesirable distortions [31]. To solve these issues, researchers have proposed generative models. This method aims to learn the underlying distribution of the target speaker's voice and use this knowledge to generate the clean speech of the target speaker from a mixture of voices rather than directly mapping from mixed speech to clean speech. Some generative models, like diffusion models [19] and variational autoencoders (VAE) [32] have been studied. It has been demonstrated that generative models can achieve results comparable to those of discriminative models [10, 19].

The modeling of discretization of audio has gained significant attention with the advancement of language models (LMs) in modeling texts [3, 8, 23, 24, 29].
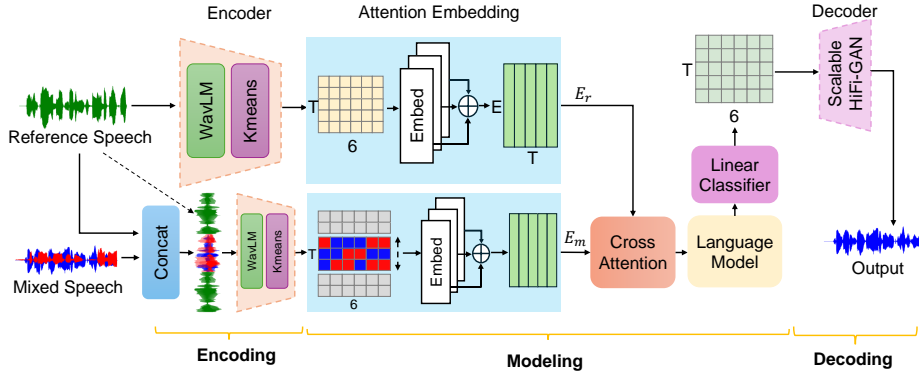
Fig. 1. Overview of our proposed target speaker extraction framework with discrete tokens and language models.

Researchers are trying to convert audio into discrete tokens and leverage LMs to model them, thereby simplifying audio generation tasks by transforming complex regression problems into classification tasks [17]. LMs have demonstrated scalability and unification of different modalities in audio synthesis tasks [4,7,14]. It has also achieved superior performances compared with discriminative models in tasks such as speech enhancement [15,34] and blind speech separation [10].

Some discretization methods [10,17,34] use Kmeans on the output of Self-Supervised Learning (SSL) models such as HuBERT [13] and WavLM [5]. SSL models have demonstrated outstanding performances across numerous downstream tasks [35], as they extract continuous representations rich in semantic and timbral information from speech. In addition, some works [9] use neural audio codec for discretization. However, their performances on Speech Separation are limited [17].

There are only few works about discretization in Target Speech Extraction (TSE). SkiM-UniCATS [36] is among the first to use discrete tokens in TSE, leveraging SSL models like HuBERT and vq-wav2vec [2]. However, it overlooks the WavLM model, which excels in speech separation [5]. Also, it mainly focuses on single-layer outputs for discretization. Moreover, its evaluation is limited to speech quality, neglecting intelligibility and speaker similarity, both crucial for TSE [28]. We try to address these potential issues by proposing TSELM, which uses multiple WavLM layers for discretization and adds metrics for intelligibility and speaker similarity. Our demos are available at [1].

2   Method

TSELM has three stages: encoding, modeling and decoding. In the encoding stage, both reference and mixed speech are tokenized using WavLM and Kmeans.
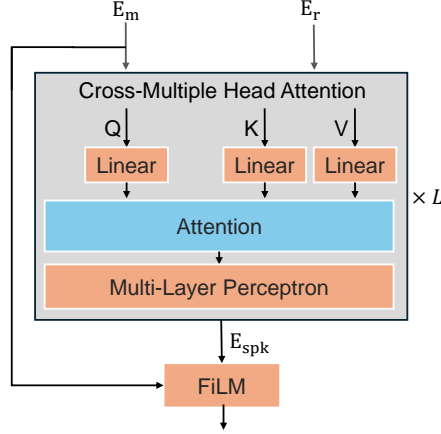
---

[1] https://beilong-tang.github.io/TSELM.demo/

Fig. 2. Details of the Cross-Attention mechanism in modeling.

The reference speech is passed directly to the encoder, while for the mixed speech, we concatenate the reference speech to both sides of the mixture before passing it through the WavLM model. After tokenization, we retain only the tokens corresponding to the mixed speech. In the modeling stage, an attention embedding mechanism is employed to combine the embeddings from all layers. A cross-attention mechanism, similar to that used in [38], is applied to inject speaker-specific information. An encoder-only language model, followed by a linear classifier, is then used to generate the reconstructed tokens. In the decoding stage, we leverage the pretrained scalable HiFi-GAN in [18] to reconstruct the audio from the discrete tokens. Unlike SELM [34], where a conformer detokenizer is trained to reconstruct WavLM embeddings before passing them through HiFi-GAN, the scalable HiFi-GAN in [18] uses a dropout mechanism to directly reconstruct audio from multiple layers of tokens, eliminating the need for a conformer detokenizer and the complexity of training a separate HiFi-GAN for each layer. Both the encoder and decoder are kept frozen during training. An overview of TSELM is shown in Fig.1. Extensive experiments demonstrate that our method achieves excellent speech quality and comparable intelligibility results.

2.1   Encoding

We use the pretrained SSL model WavLM Large [5] to encode speech into continuous representations. Specifically, we extract the outputs from six hidden layers: 1, 3, 7, 12, 18, and 23, following [17]. Given a speech signal $s \in \mathbb{R}^{T'}$, the output of WavLM is a tensor r with shape $n \times T \times E$, where n is the number of output layers (6 in this case), T is the time dimension, and E represents the embedding dimension. For tokenization, we apply a separate Kmeans model to each output layer, with each model using the same number of clusters, denoted by K. After

tokenization, the continuous embedding r is transformed into a discrete tensor d with shape $n \times T$, where each value $d_i \in (0, K-1)$. In all our experiments, we set $K = 1000$. For both reference and mixed speech, the same Kmeans model and the same layer combination from WavLM Large are used. The encoder remains frozen during training.

The encoding strategy for mixed speech is crucial to the performance of the model. Given a reference speech $s_r \in \mathbb{R}^{T^r}$ and a mixed speech $s_m \in \mathbb{R}^{T'}$, we follow the previously described procedure to encode the reference speech into a tensor d_r of shape $n \times T_r$. However, for the mixed speech, instead of applying the encoding directly, we first concatenate it with the reference speech, creating a signal $s' = [s_r, s_m, s_r] \in \mathbb{R}^{(T^r + T' + T^r)}$. This concatenated signal is then input into the encoder, producing an output tensor d' with shape $n \times (T_r + T + T_r)$, where T represents the length for the mixed speech embedding. The tensor d' contains discrete tokens for the two segments of reference speech and the mixed speech. We extract the portion d corresponding to the mixed speech, resulting in an output tensor of shape $n \times T$.

This approach is inspired by WavLM [5], which trains the model by overlapping clean speech with an interfering signal covering less than 50% of the speech, using the first utterance as the primary speaker. This allows WavLM to focus on producing target speaker dependent embeddings. Our experiments demonstrate that this concatenation strategy significantly enhances the model's performance by guiding it to prioritize the target speaker's information.

2.2   Modeling

Attention Embedding After obtaining the discrete tensor d with shape $6 \times T$, we use 6 learnable embedding tables each with K entires to embed the 6 layers respectively, each resulting in a tensor of shape $T \times E$. After embedding, we follow the same recipe as in [17] to aggregate the tensor by using attention mechanism to sum all the 6 tensors. This summation keeps the information of each layer while reducing the dimension of layers and system complexity. After attention embedding, we obtain reference embedding E_r and mixture embedding E_m.

Cross Attention We apply cross-attention module in [38] to inject the reference embeddings into the mixture. The details are shown in Fig.2. The cross-attention module consists of a stack of cross-multiple head attention modules, followed by a Feature-wise Linear Modulation (FiLM) module. We use E_m as the query and E_r as the key and value for the attention module. The output from the cross-multiple head attention module E_{spk} is passed together with E_m to the FiLM to obtain the final output. The output of FiLM $E_f = FiLM(E_m, E_{spk}) = \gamma E_{spk} \cdot E_m + \beta E_{spk}$ where γ and β are learnable parameters denoting the scaling and shifting vectors respectively.

Language Modeling We use encoder-only transformers containing multiple self-attention modules to model the embedding E_f. Due to encoder-only style,

the LM is able to learn from all the positions. Finally, 6 linear classifiers each with dimension K is used to produce the logit scores of the tokens. Cross-entropy loss is applied between the predicted tokens and the clean tokens, which are obtained by discretizing the ground truth clean audio.

3   Experiments Setup

3.1   Training

We use the publicly available Kmeans tokenizer and scalable HiFi-GAN decoder in [25]. The Kmeans tokenizer is trained on `train_clean_[100,360,500]` of LibriSpeech [21], and the scalable HiFi-GAN is trained on `train_clean_100` of LibriTTS [37]. The modeling stage is trained on `train_clean_[100,360]` of LibriSpeech. All training data are generated on the fly with relative SNR between 0 to 5 dB. The mixture audio and reference audio is clipped to 3 and 4 seconds, respectively.

We utilize the output from hidden layers 1, 3, 7, 12, 18, 23 from WavLM Large and Kmeans model with $K = 1000$. We use 1024 as the embedding dimension. The cross-attention module consists of 4 transformer encoders, each with 16 attention heads and an MLP with a hidden dimension of 1024. Layer normalization is applied after the cross-attention module. The LM of small version TSELM-S uses embedding dimension $d = 256$, absolute sinusoidal positional embedding, conformer encoders as the backbone of LM. The conformer encoder consists of 6 layers with a kernel size of 31, each with 4 attention heads, and an MLP with a hidden dimension of 2048. The medium version TSELM-M uses $d = 512$ with 8 layers and 8 heads and the large version TSELM-L uses $d = 768$ with 12 layers and 16 heads. We use AdamW as our optimizer for all the experiments. The learning rate is 5×10^{-4} for TSELM-S and 5×10^{-5} for TSELM-M and TSELM-L. We train our models using 8 GPUs with 32 GB of RAM, each with batch size 16 for a total of 40,000 steps.

3.2   Evaluation

We evaluate our models using Libri2Mix [6] and WSJ0-2mix [12]. We follow the recipes[2] to form the reference speeches for WSJ0-2mix. We use the clean testset of Libri2Mix[3], and the reference speeches are randomly selected. It is shown that metrics like PESQ, SI-SNR, STOI do not accurately reflect the speech quality of vocoder outputs due to the fact that the vocoder output does not focus strictly on frame alignment [10,34]. Instead, we use DNSMOS [26] to measure the speech quality, and the differential Word Error Rate (dWER) [33] using the base model of Whisper [22] to measure the speech intelligibility. For speaker similarity, we use the `ResNet221_LM` from the public WeSpeaker [30] to calculate the embedding cosine speaker similarity.

---

[2] https://github.com/xuchenglin28/speaker_extraction
[3] https://github.com/JorisCos/LibriMix

3.3   Baseline models

Baseline models are presented in Table 1 and 2. TSELM is compared with Spex+ [11], a discriminative separation model trained on Libri2Mix [6]. We also compare our model with SkiM-UniCATS(vq-wav2vec) from [36], which uses discrete tokens from vq-wav2vec to conduct TSE. Mixture refers to the unprocessed mixed speech. Target-Discrete refers to the discretized target speech using our encoder and decoder. It serves as the upper bond for our model performance. Besides TSELM, we conduct 3 main experiments, named Continuous-WavLM-L6, TSELM-L-Hybrid and TSELM-S-NoCat using the same training data. For Continuous-WavLM-L6, we directly pass the embeddings from the 6th hidden layer output of WavLM Large to the cross-attention without discretization. The concatenation strategy is still applied to the mixed speech. Mean Square Error loss is applied between the output embeddings and the clean embeddings. HiFi-GAN in [1] is used for audio reconstruction. For TSELM-L-Hybrid, inspired by MaskSR [15], we discretize the reference speech while utilizing the continuous embeddings from the mixed speech. In TSELM-S-NoCat, we utilize the TSELM-S model architecture but abandon the mixture concatenation strategy in the encoding step. The code and pretrained models are available at [4].

Table 1. Results on **Libri2Mix clean**. *In the "Category" column, "G" refers to generative models, while "D" refers to discriminative models. The "Type" column categorizes methods as "D" (discrete), "H" (hybrid), or "C" (continuous). For discrete methods, speaker similarity is compared against the discretized target speech (denoted as Target-Discrete) instead of the original target speech. It is denoted as "_d".*

System	Category	Type	DNSMOS ↑			dWER ↓	Spk Sim ↑
			SIG	BAK	OVL		
Mixture	-	-	3.38	3.10	2.65	79.2	0.762
Target-Discrete	G	-	3.47	4.03	3.19	11.8	0.654
Spex+ [11]	D	-	3.38	3.77	3.00	19.0	0.922
SkiM-UniCATS(vq-wav2vec) [36]	G	D	-	-	-	-	-
Continuous-WavLM-L6	G	C	3.57	4.06	3.28	14.6	0.870
TSELM-L-Hybrid	G	H	3.49	4.05	3.22	20.0	0.917_d
TSELM-S-NoCat	G	D	3.48	4.02	3.19	71.5	0.854_d
TSELM-S	G	D	3.50	4.06	3.23	28.1	0.883_d
TSELM-M	G	D	3.49	4.04	3.21	29.0	0.892_d
TSELM-L	G	D	3.49	4.04	3.21	27.5	0.895_d

---

[4] https://github.com/Beilong-Tang/TSELM

Table 2. Results on **WSJ0-2mix**.

System	Category	Type	DNSMOS ↑			dWER ↓	Spk Sim ↑
			SIG	BAK	OVL		
Mixture	-	-	3.42	3.28	2.81	63.6	-
Target-Discrete	G	-	3.56	4.09	3.30	10.1	0.657
Spex+ [11]	D	-	3.49	4.00	3.21	15.0	0.943
SkiM-UniCATS(vq-wav2vec) [36]	G	D	3.62	4.10	3.37	-	-
Continuous-WavLM-L6	G	C	3.61	4.08	3.35	8.0	0.892
TSELM-L-Hybrid	G	H	3.57	4.10	3.31	12.6	0.915_d
TSELM-S-NoCat	G	D	3.55	4.08	3.28	64.5	0.888_d
TSELM-S	G	D	3.57	4.10	3.32	19.4	0.915_d
TSELM-M	G	D	3.57	4.10	3.32	18.8	0.921_d
TSELM-L	G	D	3.57	4.10	3.31	17.8	0.924_d

4 Results and Discussions

Table 1 and 2 present the performance of different systems evaluated on the Libri2Mix testset and WSJ0-2mix testset. DNSMOS is computed over the output since this metric is reference-free. dWER is calculated with respect to the clean speech. For continuous methods, speaker similarity is directly computed against the clean speech. However, since discretization inherently results in some loss of speaker information—Target-Discrete shows a speaker similarity score of 0.654 on Libri2Mix —we follow [17] to assess the speaker similarity of the output from the discrete methods against the target audio produced by discretizing the clean speech. The observed speaker information loss is likely due to the tokenization process, which inherently reduces speaker fidelity. Future work should aim to enhance tokenization methods for SSL models to mitigate this loss. Since the primary goal of this research is to explore target speaker extraction using discretized information rather than to develop improved tokenization methods, we argue that comparing the speaker similarity output with discretized clean speech is a reasonable approach, as it represents an upper bound of performance.

We observe that TSELM-L outperforms Spex+ in terms of DNSMOS scores, indicating better speech quality, but performs slightly worse in dWER, suggesting lower speech intelligibility. One potential reason for this could be the discretization process applied to the mixed speech. Our Kmeans algorithm is trained on clean speech rather than mixed speech, which is advantageous for speech enhancement as it likely aids in denoising. However, when applied to mixed speech, this discretization might lead to the model focusing on the wrong speaker, as it is most likely to retain only the dominant speaker information, causing a reduction in intelligibility. This hypothesis is supported by the results of TSELM-L-Hybrid, where continuous embeddings from the mixture speech

Table 3. Performance of different SSL models and layer selections on Libri2Mix clean. WavLM-L6 uses only the 6th layer of hidden output of the WavLM Large model. WavLM denotes our TSELM-L model which uses 6 hidden layers.

SSL-Model	Type	DNSMOS ↑			dWER ↓	Spk Sim ↑
		SIG	BAK	OVL		
HuBERT	Discrete	3.57	4.09	3.31	82.2	0.854_d
	Hybrid	3.57	4.10	3.32	36.1	0.900_d
WavLM-L6	Discrete	2.08	2.07	1.64	122.14	0.589_d
	Hybrid	3.54	3.93	3.18	29.3	0.838_d
WavLM	Discrete	3.49	4.04	3.21	27.5	0.895_d
	Hybrid	3.49	4.05	3.22	20.0	0.917_d

are used without discretization, achieving dWER scores better than Spex+ on WSJ0-2mix.

We observe a significant increase in dWER when the mixture audio is not concatenated with the reference in the encoding step, as seen in the TSELM-S-NoCat results in Table 1 and 2. For WavLM to effectively perform target speaker separation, the input audio must follow specific conditions: the mixture should be less than 50% of the total length, and the first utterance should be the target speaker. Under these conditions, WavLM outputs a slightly denoised embedding that emphasizes the target speaker. When the entire input is a mixture, however, we found that WavLM sometimes extracts the wrong speaker. Our current concatenation strategy, inspired by SELM's [34] success in speech denoising, reframes the target speaker extraction task as a more challenging speech enhancement problem, utilizing WavLM's denoising capabilities. However, this approach remains suboptimal, and we believe our future work should prioritize developing a speaker-aware tokenization method.

In Table 3, we compare the performance of HuBERT and WavLM as SSL models and examine the effects of using either one or multiple layers for discretization. Our findings indicate that using HuBERT as the SSL model results in slightly better DNSMOS scores but much worse dWER compared to our WavLM baseline. The improved DNSMOS scores likely stem from the vocoder performance, yet they do not adequately reflect speech intelligibility, which is crucial for speech separation tasks. The poorer dWER scores observed with this HuBERT model may be attributed to its training on clean speech, which might not equip it to capture the complexity and richness of mixed speech. Moreover, our results from WavLM-L6 suggest that when performing speech separation, discretizing across multiple layers provides better results than relying on a single layer. This might be because that using multiple layer outputs can better tolerate errors compared to using just one layer output.

Finally, we observe a performance gap between discrete methods and continuous methods, as demonstrated by Continuous-WavLM-L6. Continuous-WavLM-L6 has the best performance in terms of DNSMOS and dWER among all the

experiments while using only the 6th layer output of WavLM Large. The gap in performance may be attributed to the information loss inherent in the discretization process. We hope our future research will bridge this gap.

5   Conclusion

In this work, we introduce a novel way using discrete tokens and language models for target speaker extraction. Our method leverages multiple hidden layers of WavLM and Kmeans tokenizers for encoding, employs cross-attention and a language model for separation, and utilizes a scalable HiFi-GAN for audio reconstruction. Experiments have shown that our model can achieve excellent performance in terms of speech quality, and comparable performance in terms of speech intelligibility. However, we observe a gap between discrete methods and continuous methods especially in speech intelligibility and speaker similarity. Our future work will focus on shrinking this gap.

6   Acknowledgment

References

1. Baas, M., van Niekerk, B., Kamper, H.: Voice conversion with just nearest neighbors. In: Interspeech (2023)
2. Baevski, A., Schneider, S., Auli, M.: vq-wav2vec: Self-supervised learning of discrete speech representations. arXiv preprint arXiv:1910.05453 (2019)
3. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. In: Advances in neural information processing systems. vol. 33, pp. 1877–1901 (2020)
4. Chen, Q., Chu, Y., Gao, Z., Li, Z., Hu, K., Zhou, X., Xu, J., Ma, Z., Wang, W., Zheng, S., et al.: Lauragpt: Listen, attend, understand, and regenerate audio with gpt. arXiv preprint arXiv:2310.04673 (2023)
5. Chen, S., Wang, C., Chen, Z., Wu, Y., Liu, S., Chen, Z., Li, J., Kanda, N., Yoshioka, T., Xiao, X., Wu, J., Zhou, L., Ren, S., Qian, Y., Qian, Y., Wu, J., Zeng, M., Yu, X., Wei, F.: Wavlm: Large-scale self-supervised pre-training for full stack speech processing. IEEE Journal of Selected Topics in Signal Processing **16**(6), 1505–1518 (2022). https://doi.org/10.1109/JSTSP.2022.3188113
6. Cosentino, J., Pariente, M., Cornell, S., Deleforge, A., Vincent, E.: Librimix: An open-source dataset for generalizable speech separation. arXiv preprint arXiv:2005.11262 (2020)
7. Deshmukh, S., Elizalde, B., Singh, R., Wang, H.: Pengi: An audio language model for audio tasks. Advances in Neural Information Processing Systems **36**, 18090–18108 (2023)

8. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
9. Du, Z., Wang, J., Chen, Q., Chu, Y., Gao, Z., Li, Z., Hu, K., Zhou, X., Xu, J., Ma, Z., et al.: Lauragpt: Listen, attend, understand, and regenerate audio with gpt. arXiv preprint arXiv:2310.04673 (2023)
10. Erdogan, H., Wisdom, S., Chang, X., Borsos, Z., Tagliasacchi, M., Zeghidour, N., Hershey, J.R.: Tokensplit: Using discrete speech representations for direct, refined, and transcript-conditioned speech separation and recognition. arXiv preprint arXiv:2308.10415 (2023)
11. Ge, M., Xu, C., Wang, L., Chng, E.S., Dang, J., Li, H.: Spex+: A complete time domain speaker extraction network. arXiv preprint arXiv:2005.04686 (2020)
12. Hershey, J.R., Chen, Z., Le Roux, J., Watanabe, S.: Deep clustering: Discriminative embeddings for segmentation and separation. In: 2016 IEEE international conference on acoustics, speech and signal processing (ICASSP). pp. 31–35. IEEE (2016)
13. Hsu, W.N., Bolte, B., Tsai, Y.H.H., Lakhotia, K., Salakhutdinov, R., Mohamed, A.: Hubert: Self-supervised speech representation learning by masked prediction of hidden units. IEEE/ACM transactions on audio, speech, and language processing **29**, 3451–3460 (2021)
14. Kreuk, F., Synnaeve, G., Polyak, A., Singer, U., Défossez, A., Copet, J., Parikh, D., Taigman, Y., Adi, Y.: Audiogen: Textually guided audio generation. arXiv preprint arXiv:2209.15352 (2022)
15. Li, X., Wang, Q., Liu, X.: Masksr: Masked language model for full-band speech restoration. arXiv preprint arXiv:2406.02092 (2024)
16. Luo, Y., Mesgarani, N.: Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation. IEEE/ACM transactions on audio, speech, and language processing **27**(8), 1256–1266 (2019)
17. Mousavi, P., Della Libera, L., Duret, J., Ploujnikov, A., Subakan, C., Ravanelli, M.: Dasb–discrete audio and speech benchmark. arXiv preprint arXiv:2406.14294 (2024)
18. Mousavi, P., Duret, J., Zaiem, S., Della Libera, L., Ploujnikov, A., Subakan, C., Ravanelli, M.: How should we extract discrete audio tokens from self-supervised models? arXiv preprint arXiv:2406.10735 (2024)
19. Nguyen, T., Sun, G., Zheng, X., Zhang, C., Woodland, P.C.: Conditional diffusion model for target speaker extraction. arXiv preprint arXiv:2310.04791 (2023)
20. Pal, M., Roy, R., Basu, J., Bepari, M.S.: Blind source separation: A review and analysis. In: 2013 International Conference Oriental COCOSDA held jointly with 2013 Conference on Asian Spoken Language Research and Evaluation (O-COCOSDA/CASLRE). pp. 1–5. IEEE (2013)
21. Panayotov, V., Chen, G., Povey, D., Khudanpur, S.: Librispeech: an asr corpus based on public domain audio books. In: 2015 IEEE international conference on acoustics, speech and signal processing (ICASSP). pp. 5206–5210. IEEE (2015)
22. Radford, A., Kim, J.W., Xu, T., Brockman, G., McLeavey, C., Sutskever, I.: Robust speech recognition via large-scale weak supervision. In: International conference on machine learning. pp. 28492–28518. PMLR (2023)
23. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I.: Improving language understanding by generative pre-training. OpenAI Blog **1**(8) (2018)
24. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research **21**(140), 1–67 (2020)

25. Ravanelli, M., Parcollet, T., Plantinga, P., Rouhe, A., Cornell, S., Lugosch, L., Subakan, C., Dawalatabad, N., Heba, A., Zhong, J., et al.: Speechbrain: A general-purpose speech toolkit. arXiv preprint arXiv:2106.04624 (2021)

26. Reddy, C.K., Gopal, V., Cutler, R.: Dnsmos p. 835: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors. In: ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 886–890. IEEE (2022)

27. Subakan, C., Ravanelli, M., Cornell, S., Bronzi, M., Zhong, J.: Attention is all you need in speech separation. In: ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 21–25 (2021). `https://doi.org/10.1109/ICASSP39728.2021.9413901`

28. Tang, B., Zeng, B., Li, M.: Tselm: Target speaker extraction using discrete tokens and language models. arXiv preprint arXiv:2409.07841 (2024)

29. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)

30. Wang, H., Liang, C., Wang, S., Chen, Z., Zhang, B., Xiang, X., Deng, Y., Qian, Y.: Wespeaker: A research and production oriented speaker embedding learning toolkit. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 1–5. IEEE (2023)

31. Wang, P., Tan, K., et al.: Bridging the gap between monaural speech enhancement and recognition with distortion-independent acoustic modeling. IEEE/ACM Transactions on Audio, Speech, and Language Processing **28**, 39–48 (2019)

32. Wang, R., Li, L., Toda, T.: Dual-channel target speaker extraction based on conditional variational autoencoder and directional information. IEEE/ACM Transactions on Audio, Speech, and Language Processing **32**, 1968–1979 (2024). `https://doi.org/10.1109/TASLP.2024.3376154`

33. Wang, Z.Q., Erdogan, H., Wisdom, S., Wilson, K., Raj, D., Watanabe, S., Chen, Z., Hershey, J.R.: Sequential multi-frame neural beamforming for speech separation and enhancement. In: 2021 IEEE Spoken Language Technology Workshop (SLT). pp. 905–911. IEEE (2021)

34. Wang, Z., Zhu, X., Zhang, Z., Lv, Y., Jiang, N., Zhao, G., Xie, L.: Selm: Speech enhancement using discrete tokens and language models. In: ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 11561–11565. IEEE (2024)

35. Yang, S.w., Chi, P.H., Chuang, Y.S., Lai, C.I.J., Lakhotia, K., Lin, Y.Y., Liu, A.T., Shi, J., Chang, X., Lin, G.T., et al.: Superb: Speech processing universal performance benchmark. arXiv preprint arXiv:2105.01051 (2021)

36. Yu, L., Zhang, W., Du, C., Zhang, L., Liang, Z., Qian, Y.: Generation-based target speech extraction with speech discretization and vocoder. In: ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 12612–12616 (2024). `https://doi.org/10.1109/ICASSP48485.2024.10446418`

37. Zen, H., Dang, V., Clark, R., Zhang, Y., Weiss, R.J., Jia, Y., Chen, Z., Wu, Y.: Libritts: A corpus derived from librispeech for text-to-speech. arXiv preprint arXiv:1904.02882 (2019)

38. Zeng, B., Li, M.: Usef-tse: Universal speaker embedding free target speaker extraction. arXiv preprint arXiv:2409.02615 (2024)

39. Zeng, B., Suo, H., Wan, Y., Li, M.: Sef-net: Speaker embedding free target speaker extraction network. In: Proc. Interspeech. pp. 3452–3456 (2023)

40. Zmolikova, K., Delcroix, M., Ochiai, T., Kinoshita, K., Černockỳ, J., Yu, D.: Neural target speech extraction: An overview. IEEE Signal Processing Magazine **40**(3), 8–29 (2023)