Interspeech 2025
17-21 August 2025, Rotterdam, The Netherlands

Neural Speech Extraction with Human Feedback

Malek Itani[1], *Ashton Graves*[1], *Sefik Emre Eskimez*[2], *Shyamnath Gollakota*[1]

[1]University of Washington, USA
[2]Sesame AI, USA

malek@cs.washington.edu, graveash@uw.edu, eeskimez@sesame.com, gshyam@cs.washington.edu

Abstract

We present the first neural target speech extraction (TSE) system that uses human feedback for iterative refinement. Our approach allows users to mark specific segments of the TSE output, generating an edit mask. The refinement system then improves the marked sections while preserving unmarked regions. Since large-scale datasets of human-marked errors are difficult to collect, we generate synthetic datasets using various automated masking functions and train models on each. Evaluations show that models trained with noise power-based masking (in dBFS) and probabilistic thresholding perform best, aligning with human annotations. In a study with 22 participants, users showed a preference for refined outputs over baseline TSE. Our findings demonstrate that human-in-the-loop refinement is a promising approach for improving the performance of neural speech extraction.

Index Terms: Source separation, human-in-the-loop

1. Introduction

Despite advancements in model architectures and training techniques [1, 2, 3, 4], neural speech extraction remains an unsolved problem, with no approach achieving consistently robust performance. Target speech extraction (TSE) models struggle to extract the target speaker when speech overlaps, the enrollment signal differs in acoustic characteristics from the mixture, or interfering speakers have similar vocal traits [5]. Unlike background noise separation, distinguishing between subtle differences in human voices is far more complex [6]. Consequently, TSE models may make mistakes in certain segments or incorrectly identify the speaker in some or all parts of the output.

Here, we present the first neural TSE system that incorporates human feedback for iterative refinement. As shown in Fig. 1, the system processes an input mixture using a TSE model to extract speech. If the output is unsatisfactory, the user can provide feedback by marking specific segments to generate an edit mask. Our refinement system then utilizes the initial extraction and the edit mask to enhance the speech signal, modifying only the designated sections while preserving unaltered regions.

Human feedback has been helpful for aligning text generated by large language models with human preferences [7] and has also been used to guide image editing to meet user needs [8, 9]. However, building a refinement network for TSE using human feedback is challenging due to limited datasets.

Collecting large-scale datasets of human-marked errors in neural speech extraction output is not feasible. Instead, we create synthetic datasets that approximate human feedback. Specifically, we generate multiple synthetic datasets using various masking functions and train separate refinement models on each. These models are then evaluated on a set of 200 audio samples, annotated with edit masks from human annotators. Our results show that the model trained with noise power-based masking function (in dBFS) with probabilistic thresholding yields the best improvements on human-annotated samples. This aligns with human perception of loudness being roughly on a logarithmic scale [10], while probabilistic thresholding accounts for variations in how humans create masks; making our synthetic data more representative of human feedback.

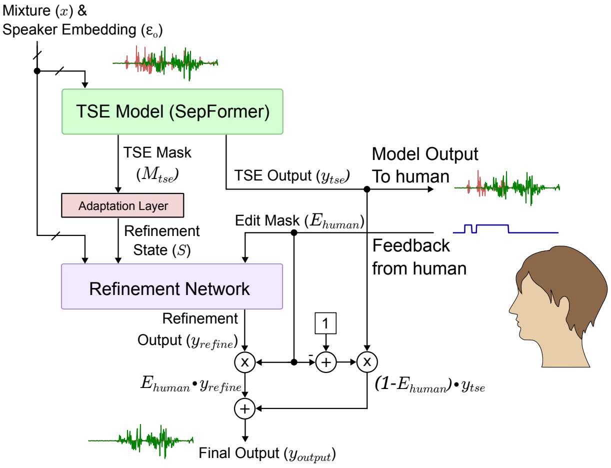

Figure 1: *System architecture for human-in-the-loop TSE. A neural TSE model extracts an initial target speech estimate, y_{tse}, which a human reviews and marks for refinement. Our proposed refinement model then incorporates this human feedback to produce a more refined target speech estimate.*

We develop an interactive tool that lets users listen to TSE model outputs and mark regions for refinement. To validate our system, we recruit 22 additional participants (10 annotators, 12 listeners) with varying audio processing backgrounds. The annotators create edit masks for another 200 samples, and listeners rate the audio quality before and after refinement. Mean opinion scores reveal that participants prefer the refined audio from human feedback over the TSE-only output.

2. Related work

Target speech extraction: This task aims to extract a target speaker from a mixture using cues such as audio examples [3, 1, 2, 5, 11, 6], spatial [12, 13, 14], visual [15], text [16], or concept embeddings [17]. While these works proposed various architectures to improve performance [4, 18, 19], our work is complementary in that it addresses the imperfections of neural networks by integrating human feedback at inference time.

Audio editing. Pre-deep learning audio editing tools, such

10.21437/Interspeech.2025-214

as [20], enabled users to separate audio sources from a mixture by painting on time-frequency visualizations. More recent approaches employed transformers [21] and diffusion models [22] to enable modifications in both audio mixtures and music [23]. These methods leveraged text-based [24] and instruction-guided methods [25] to enable precise control over musical features like chords and rhythm [21] as well as replaced audio classes in mixtures [25]. More recent work [23] demonstrated editing of specific audio features, including speaker pitch, duration, volume, and spectral balance. However, none of these approaches addressed the challenge of multiple speech sources in a mixture or the task of target speech extraction.

Dynamic inference. Prior work explored dynamic inference for speech separation using purely computational strategies [26]. Slimmable neural networks adjust the width of the network at run-time [26, 27] while early exit methods [28, 29] halt computation based on prediction similarity or gating decisions. In contrast, our approach integrates human feedback into neural speech extraction, allowing users to provide edit masks to refine the quality of the speech extraction. Our human-in-the-loop approach is complementary to existing deep learning-only strategies, enabling more accurate outputs.

3. Methods

Problem Formulation. Let $x \in \mathbb{R}^T$ be a noisy recording containing speech from a target speaker s_0, mixed with K interfering speakers s_i $(i = 1, \ldots, K)$ and noise n.

$$x = s_0 + \sum_{i=1}^{K} s_i + n \qquad (1)$$

The goal of neural TSE network, $\mathcal{F}$, is to extract an estimate s_0^{tse} of the target speech from x, using a speaker embedding ε_0:

$$y_{tse} = s_0^{tse} = \mathcal{F}(x|\varepsilon_0) \qquad (2)$$

Since $\mathcal{F}$ may not always produce an accurate estimate of the target speech, we seek to design a refinement network $\mathcal{G}$ that uses human feedback about the TSE output, s_0^{tse}. The feedback can be provided in the form of a binary edit mask, $E_{human} \in \mathbb{Z}_2^T$, where $E_{human}[i] = 1$ if the user marks the i-th sample for refinement and $E_{human}[i] = 0$ otherwise. The goal is to obtain a refined estimate $s_0^{refined}$ that better approximates s_0:

$$y_{refine} = s_0^{refined} = \mathcal{G}(x|\varepsilon_0; E_{human}; s_0^{tse}) \qquad (3)$$

System Architecture. Fig. 1 shows our human-in-the-loop neural TSE system. The TSE network is based on SepFormer [30], a transformer-based architecture. We condition the model on the target speaker using a FiLM [31] layer, which is applied after the SepFormer encoder. We use d-vectors [2] to condition the network on the target speaker characteristics.

Formally, the TSE model encodes x into a latent representation with C_{tse} channels and T' time steps. It then generates a mask $M_{tse} \in \mathbb{R}^{C_{tse} \times T'}$, which is applied to the encoded audio representation. A learned decoder reconstructs the estimated time-domain target speech signal, $y_{tse} \in \mathbb{R}^T$.

The output of the TSE network is presented to the user, who provides feedback by marking samples that need refinement. Once the user provides the edit mask, M_{tse} is transformed into a refinement state $S \in \mathbb{R}^{C_R \times T'}$ using a fully-connected adaptation layer, where C_R is the number of refinement channels. The refinement network then incorporates S and the edit mask to generate the refined target speech output $y_{refine} \in \mathbb{R}^T$. We obtain the final estimate $y_{output} \in \mathbb{R}^T$ by updating only the sections marked for refinement in the edit mask:

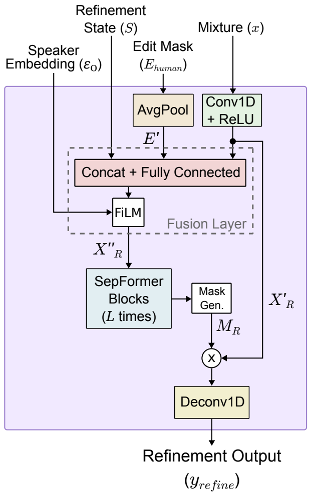

Figure 2: *Refinement network architecture. The encoded mixture, downsampled edit mask, TSE refinement state, and speaker embedding are fused into a conditioned tensor and processed by SepFormer blocks and a mask generator to produce a mask.*

$$y_{output} = E_{human} \cdot y_{refine} + (1 - E_{human}) \cdot y_{tse} \qquad (4)$$

Refinement Network. Fig. 2 shows how our network integrates human feedback to generate a more accurate approximation of the target speech. Similar to the TSE model, we first encode the mixture x into a latent representation $X'_R \in \mathbb{R}^{C_R \times T'}$ using a strided convolution with a ReLU activation. This encoder doesn't share parameters with the TSE model.

To align the edit mask E_{human} with the temporal resolution of X'_R, we apply average pooling to obtain a downsampled tensor E' with the same number of time steps as X'_R. Then, we fuse X'_R, E', the refinement state S, and ε_0 with a fusion layer. This layer performs channel concatenation, followed by a fully-connected layer, and finally, a FiLM layer to condition the input on the edit mask, refinement state, and speaker embedding. This fusion process produces a conditioned tensor $X''_R \in \mathbb{R}^{C_R \times T'}$.

We then pass X''_R through a series of L SepFormer blocks, followed by a fully-connected mask generator to produce a refinement mask $M_R \in \mathbb{R}^{C_R \times T'}$. Finally, we multiply this mask with X'_R and use a deconvolution layer to decode the refined target speech output $y_{refine} \in \mathbb{R}^T$.

Automated Masking Functions as Substitutes for Human-Generated Masks. To learn $\mathcal{G}$ using a data-driven approach, we require edit masks that capture human evaluations of TSE model outputs. However, collecting a sufficiently large dataset of human-annotated edit masks is impractical. Instead, we approximate them using masking functions.

Since the clean target speech s_0 represents an ideal reference, we assume that humans perceive deviations from s_0 in s_0^{tse} as noise. When this deviation surpasses a certain threshold, a user would likely mark the region. Thus, our masking function quantifies the dissimilarity between s_0 and s_0^{tse}, assigning ones to regions exceeding the threshold and zeros otherwise.

Formally, we define the masking function as a mapping $f(A, B) : \mathbb{R}^N \times \mathbb{R}^N \rightarrow \mathbb{Z}_2^N$, where the synthetic edit mask is computed as $E_{synthetic} = f(s_0^{tse}, s_0)$. Our goal is to choose a masking function such that $E_{synthetic}$ closely aligns with human annotations, i.e., $E_{synthetic} \sim E_{human}$.

Table 1: *Comparing model configurations and masking functions. TSE+Refine is our proposed refinement strategy, while TSE+TSE successively applies the same TSE model whenever a masking function has any non-zero sample.*

Config.	**Masking function**	**Count**	**SISDR (dB)**	**PESQ**	**DNSMOS OVRL**
Mixture	–	–	0.01	1.26	2.48
TSE	–	–	12.18	1.91	3.21
TSE+Refine	meanAE	109	12.92	1.92	3.22
TSE+TSE	meanAE	109	11.92	1.91	3.21
TSE+Refine	maxAE	798	14.03	2.03	3.37
TSE+TSE	maxAE	798	9.72	1.80	3.12
TSE+Refine	GlobalSNR	41	12.72	1.93	3.23
TSE+TSE	GlobalSNR	41	12.09	1.91	3.21
TSE+Refine	dBFS	945	14.07	2.07	3.40
TSE+TSE	dBFS	945	9.18	1.76	3.08
TSE+Refine	dBFS-prob	917	**14.88**	**2.16**	**3.49**
TSE+TSE	dBFS-prob	917	9.25	1.77	3.09

Table 2: *TSE+Refinement results using different masking functions with 2-speaker VCTK mixtures using human annotations.*

Config/ Masking function	**SI-SDR (dB)**	**PESQ**	**DNSMOS OVRL**
TSE	-0.93	1.60	2.72
TSE+Refine			
meanAE	2.40	1.72	3.05
maxAE	4.75	1.83	3.00
GlobalSNR	-5.73	1.61	**3.07**
dBFS	4.57	1.80	2.99
dBFS-prob	**5.76**	**1.85**	3.02

To compute the edit mask, we segment s_0 and s_0^{tse} into non-overlapping 0.25-second windows, i.e., $N = 4000$ at a sampling rate of 16 kHz, and calculate the masking function between pairs of windows corresponding to the same segment in time, and concatenate the results. This produces a fine-grained edit mask that varies over time. Additionally, we also define a global edit mask, which applies a single value across the entire signal, either marking all or none of it for refinement.

Masking functions have the following form:

$$f(A, B) = \begin{cases} \mathbf{1}^N & \text{if } g(A, B) > \tau \\ \mathbf{0}^N & \text{otherwise} \end{cases}$$

where $g : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}$ is a similarity metric and τ is a threshold value. In this work, we look at five different masking functions, which differ in the choice of g and τ:

- Fine-grained Mean Absolute Error (*meanAE*): $g(A, B) = \frac{1}{N} \sum_i |A_i - B_i|$; $\tau = 0.03$
- Fine-grained Max Absolute Error (*maxAE*): $g(A, B) = \max_i |A_i - B_i|$; $\tau = 0.1$
- Fine-grained decibels relative to full-scale (*dBFS*): $g(A, B) = 10 \log \frac{1}{N} \sum_i (A_i - B_i)^2$; $\tau = -40$
- Fine-grained dBFS, probabilistic (*dBFS-prob*): $g(A, B) = 10 \log \frac{1}{N} \sum_i (A_i - B_i)^2$; $\tau \sim \mathcal{N}(-40, \sigma = 3)$
- Global signal-to-noise ratio (*GlobalSNR*): $g(A, B) = -\text{SNR}(A, B)$; $\tau = -5$

$\text{SNR}(\hat{x}, x) = 10 \log_{10} \left(\frac{||x||_2^2}{||x - \hat{x}||_2^2} \right)$ is the signal-to-noise ratio.

Table 3: *Evaluation on 2-speaker VCTK mixtures.*

Config.	**SI-SDR (dB)**	**PESQ**	**DNSMOS**
Mixture	-0.16	1.54	2.98
TSE	10.81	2.02	3.07
TSE+Refine	**13.08**	**2.22**	**3.28**

Table 4: *Evaluation on 3-speaker LibriSpeech mixtures.*

Config.	**SI-SDR (dB)**	**PESQ**	**DNSMOS**
Mixture	-0.3	1.16	2.14
TSE	8.21	1.51	2.81
TSE+Refine	**10.62**	**1.65**	**3.06**

4. Experiments and Results

Datasets. We trained our models on 16 kHz speech from LibriSpeech [32] and noise from WHAM! [33], generating training mixtures on-the-fly via dynamic mixing. Each 5-second mixture was created by randomly selecting K speaker utterances from the same corpus split. Utterances longer than 5 seconds were cropped, and shorter ones were zero-padded with random silence. One speaker was designated as the target, with their d-vector embedding derived from a separate utterance.

Each training epoch included 20,000 mixtures, with validation and test sets fixed at 2,000 and 1,000 samples, respectively. Speech data came from LibriSpeech's `train-clean-360`, `test-clean`, and `dev-clean` splits, while noise data was sampled from WHAM!'s `tr`, `cv`, and `tt` splits. Interferer and noise amplitudes were scaled for a target speaker SNR uniformly distributed between -10 dB and 10 dB.

Training setup. We first trained the TSE model independently, then froze its weights and trained the adaptation layer and refinement network together. All models were trained for 300 epochs. For TSE models, the learning rate (LR) started at 0.002, halving after 4 epochs of no validation loss improvement. For refinement models, LR started at 0.001, halving after 6 stagnant epochs. All models used the AdamW optimizer with weight decay 0.01 and a gradient clipping of 1. The TSE model was based on SepFormer, with an encoder using a kernel size of 32 and output channel dimension $C_{tse} = 64$. SepFormer had a chunk size of 250, $L = 2$ layers, and intra-/inter-attention modules with 8 attention heads and 4 repetitions. The refinement model had the same configuration, with $C_R = 64$. The models were trained to minimize negative SI-SDR [34] between outputs and ground truth speech. While metrics were computed on the final output y_{output} during evaluation, training and validation losses were based on the refinement model output y_{refine}. Results are reported using the model weights with the lowest validation loss.

Refinement versus Successive TSE. We evaluate our refinement strategy using SI-SDR [34], PESQ [35], and personalized DNSMOS [36] across all input samples. We first assess the system by training and testing with different masking functions, measuring speech quality after refinement. Here, we focus on the two-speaker case without background noise. We also evaluate a baseline approach where samples needing refinement are passed again through the original TSE network.

Table 1 shows that repeatedly applying the TSE network degrades performance, whereas the proposed refinement method enhances target speech quality across all evaluation metrics and masking functions. This degradation likely stems from the TSE model over-suppressing the target speaker, leading to ir-

Table 5: *Evaluation on noisy 2-speaker LibriSpeech mixtures.*

Config.	SI-SDR (dB)	PESQ	DNSMOS
Mixture	-0.15	1.17	2.37
TSE	10.48	1.61	2.96
TSE+Refine	**12.27**	**1.72**	**3.22**

reversible information loss in the absence of the original mixture. Among the tested masking functions, dBFS-prob achieves the highest performance gains, improving SI-SDR, PESQ, and DNSMOS OVRL from 12.18 dB, 1.91, and 3.21 (TSE only) to 14.88 dB, 2.16, and 3.49, respectively.

The `count` variable represents the number of TSE output samples identified for refinement under each masking function. Unlike GlobalSNR, which flags only samples with an average output SNR below 5 dB over the full 5-second recording, dBFS-prob exhibits greater sensitivity to localized errors, leading to a higher number of samples selected for refinement.

Validating masking functions with human annotations. To this end, we first generated 200 TSE output samples, as our refinement procedure operates on TSE outputs. These TSE output samples were selected to ensure a uniform distribution of SI-SDR with an average SI-SDR close to 0 dB.

We developed an interactive tool (Fig. 3) that enables users to listen to and visualize the time-domain waveforms of the mixture, enrollment audio, and TSE output. Four participants were recruited to use this tool to annotate regions in the TSE output that required refinement. These annotations were then converted into edit masks, where samples were assigned a value of 1 in the marked regions and 0 elsewhere.

After collecting the human edit masks, we applied our refinement models using these masks. The results in Table 2 show that our refinement method improves speech quality. However, the choice of masking function influences its transferability to human annotations. The dBFS-prob masking function yields the best performance, improving SI-SDR by 6.69 dB over the TSE output. Additionally, it increases SI-SDR by approximately 1.2 dB over dBFS, suggesting that a probabilistic threshold can serve as an effective data augmentation strategy to capture diverse user preferences. For all subsequent evaluations, we use the model trained with dBFS-prob as the default.

Additional experiments. To evaluate our models on a different dataset, we created a test set of 1,000 two-speaker speech mixtures following the same procedure as before but using speech data from the VCTK corpus [37]. These mixtures were processed using our TSE refinement strategy trained on LibriSpeech mixtures. As shown in Table 3, our method improves the average SI-SDR over the TSE model by 2.27 dB, demonstrating its ability to generalize to out-of-distribution datasets.

We also evaluate our refinement strategy on datasets with three speakers and two speakers plus noise. For each of these scenarios, we train separate TSE and refinement models. The results are shown in Tables 4 and 5, respectively. Our refinement algorithm consistently improves all metrics in multiple background speakers and noisy scenarios. In the 3-speaker case, refinement improves the SI-SDR by 2.41 dB on average, while in the noisy speaker case, it can improve by 1.79 dB.

Finally, we replace average pooling with a 1D strided convolution (kernel size 32, stride 16, 1 output channel). Testing on the 2-speaker LibriSpeech dataset with the dBFS-prob masking function yields an SI-SDR difference within 0.06 dB.

Figure 3: *Interactive tool interface used for human evaluation.*

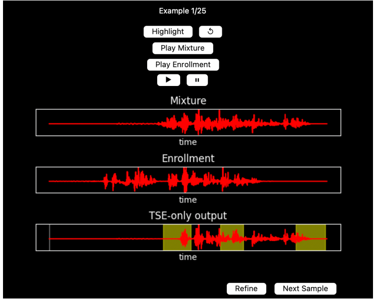

Table 6: *Objective and subjective results for human evaluation. Since refinement is useful only when TSE underperforms, samples were selected so the average TSE output SI-SDR is ∼0 dB.*

Config.	SI-SDR (dB)	PESQ	MOS
TSE	-0.55	1.59	2.10
TSE+Refine	4.79	1.80	**2.70**
TSE+Refine-replace	**4.96**	**1.85**	2.55

Human subjective evaluation. Finally, we evaluate our TSE refinement system using human-annotated edit masks from a completely new set of annotators. We created a new dataset of 200 mixtures, ensuring that the SI-SDR distribution aligned with that of the previous human evaluation. Ten additional random annotators used our tool to listen to and annotate regions for refinement. Each participant annotated 25 samples, with the first five serving as a familiarization phase and subsequently discarded. The remaining 20 annotations per participant were converted into edit masks using the same procedure described earlier. We applied our refinement algorithm to these 200 human-annotated samples and computed the objective results, shown in Table 6. Our method consistently improves SI-SDR and PESQ, demonstrating that both our approach and the selected masking function generalize effectively to unseen annotators.

To assess subjective quality, we recruited 12 additional participants to rate the quality of the the TSE output, and our refined output for a randomly selected 15 audio examples from the 200 annotated samples. These samples were presented with an enrollment audio of the target speaker. Table 6 shows that participants favored our refined output, with the mean opinion score (MOS) increasing by 0.6 points. Paired t-tests between our TSE+Refine model and TSE, and between TSE+Refine model and TSE+Refine-replace show a statistically significant difference with $p < 0.01$ and $p < 0.1$ respectively. This confirms that our refinement system enhances both objective speech quality and human-perceived audio clarity. Interestingly, while TSE+Refine-replace, which uses y_{refine} and not y_{output}, improves objective metrics, the participants preferred TSE+Refine. This may be due to the refinement model introducing subtle artifacts outside the annotated regions, impacting how user perceive the overall audio quality.

5. Conclusion

We present a neural speech extraction system incorporating human feedback for iterative refinement. Our work has limitations offering exciting future research opportunities. Focused on a single refinement iteration to minimize user effort. Exploring multi-iteration refinement networks improving performance while minimizing user effort is valuable. Our system allows marking segments for refinement, but detailed within-segment feedback (e.g., “reduce noise further”) could be explored. Finally, exploring generative models, like diffusion models, with human feedback for TSE could yield additional improvements.

Acknowledgments. The researchers are partly supported by the Moore Inventor Fellow award #10617, Thomas J. Cable Endowed Professorship, and a UW CoMotion innovation gap fund. This work was facilitated through the use of computational, storage, and networking infrastructure provided by the UW HYAK Consortium.

6. References

[1] O. Glembek, L. Burget, P. Matějka, M. Karafiát, and P. Kenny, "Simplification and optimization of i-vector extraction," in *ICASSP*, 2011.

[2] Q. Wang, C. Downey, L. Wan, P. A. Mansfield, and I. L. Moreno, "Speaker diarization with lstm," in *ICASSP*. IEEE, 2018.

[3] K. Zmolikova, M. Delcroix, T. Ochiai, K. Kinoshita, J. Černocký, and D. Yu, "Neural target speech extraction: An overview," *IEEE Signal Processing Magazine*, 2023.

[4] Z.-Q. Wang, S. Cornell, S. Choi, Y. Lee, B.-Y. Kim, and S. Watanabe, "Tf-gridnet: Making time-frequency domain models great again for monaural speaker separation," in *ICASSP*, 2023.

[5] K. Žmolíková, M. Delcroix, K. Kinoshita, T. Ochiai, T. Nakatani, L. Burget, and J. Černockỳ, "Speakerbeam: Speaker aware neural network for target speaker extraction in speech mixtures," *IEEE Journal of Selected Topics in Signal Processing*, 2019.

[6] B. Veluri, M. Itani, T. Chen, T. Yoshioka, and S. Gollakota, "Look once to hear: Target speech hearing with noisy examples," in *ACM CHI*, 2024.

[7] Y. Bai, A. Jones, K. Ndousse, A. Askell, A. Chen, N. DasSarma, D. Drain, S. Fort, D. Ganguli, T. Henighan, N. Joseph, S. Kadavath, J. Kernion, T. Conerly, S. El-Showk, N. Elhage, Z. Hatfield-Dodds, D. Hernandez, T. Hume, S. Johnston, S. Kravec, L. Lovitt, N. Nanda, C. Olsson, D. Amodei, T. Brown, J. Clark, S. McCandlish, C. Olah, B. Mann, and J. Kaplan, "Training a helpful and harmless assistant with reinforcement learning from human feedback," 2022.

[8] T. Brooks, A. Holynski, and A. A. Efros, "Instructpix2pix: Learning to follow image editing instructions," in *CVPR*, 2023.

[9] H. Ling, K. Kreis, D. Li, S. W. Kim, A. Torralba, and S. Fidler, "Editgan: High-precision semantic image editing," in *Neural Information Processing Systems*, 2021.

[10] R. Litovsky, "Chapter 3 - development of the auditory system," in *The Human Auditory System*, ser. Handbook of Clinical Neurology, M. J. Aminoff, F. Boller, and D. F. Swaab, Eds. Elsevier, 2015, vol. 129, pp. 55–72. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780444626301000032

[11] B. Veluri, J. Chan, M. Itani, T. Chen, T. Yoshioka, and S. Gollakota, "Real-time target sound extraction," in *ICASSP 2023*.

[12] A. Wang, M. Kim, H. Zhang, and S. Gollakota, "Hybrid neural networks for on-device directional hearing," *AAAI*, 2022.

[13] R. Gu, L. Chen, S.-X. Zhang, J. Zheng, Y. Xu, M. Yu, D. Su, Y. Zou, and D. Yu, "Neural spatial filter: Target speaker speech separation assisted with directional information." in *Interspeech*, 2019, pp. 4290–4294.

[14] M. Itani, T. Chen, T. Yoshioka, and S. Gollakota, "Creating speech zones with self-distributing acoustic swarms," *Nature Communications*, vol. 14, 09 2023.

[15] J. Lin, X. Cai, H. Dinkel, J. Chen, Z. Yan, Y. Wang, J. Zhang, Z. Wu, Y. Wang, and H. Meng, "Av-sepformer: Cross-attention sepformer for audio-visual target speaker extraction," in *ICASSP 2023*.

[16] X. Liu, Q. Kong, Y. Zhao, H. Liu, Y. Yuan, Y. Liu, R. Xia, Y. Wang, M. D. Plumbley, and W. Wang, "Separate anything you describe," *arXiv preprint arXiv:2308.05037*, 2023.

[17] Y. Ohishi, M. Delcroix, T. Ochiai, S. Araki, D. Takeuchi, D. Niizumi, A. Kimura, N. Harada, and K. Kashino, "Conceptbeam: Concept driven target speech extraction," ser. ACM MM, 2022.

[18] C. Subakan, M. Ravanelli, S. Cornell, F. Lepoutre, and F. Grondin, "Resource-efficient separation transformer," *arXiv preprint arXiv:2206.09507*, 2022.

[19] K. Li, G. Chen, R. Yang, and X. Hu, "Spmamba: State-space model is all you need in speech separation," 2024.

[20] N. J. Bryan, G. J. Mysore, and G. Wang, "Isse: an interactive source separation editor," in *ACM CHI*, 2014.

[21] Y.-H. Lan, W.-Y. Hsiao, H.-C. Cheng, and Y.-H. Yang, "Musicongen: Rhythm and chord control for transformer-based text-to-music generation," in *ISMIR*, 2024.

[22] M. Xu, C. Li, D. Zhang, D. Su, W. Liang, and D. Yu, "Prompt-guided precise audio editing with diffusion models," in *ICML*, 2024.

[23] M. Morrison, C. Churchwell, N. Pruyne, and B. Pardo, "Fine-grained and interpretable neural speech editing," *InterSpeech*, 2024.

[24] Y. Zhang, Y. Ikemiya, G. Xia, N. Murata, M. A. Martnez-Ramrez, W.-H. Liao, Y. Mitsufuji, and S. Dixon, "Musicmagus: Zero-shot text-to-music editing via diffusion models," in *IJCAI-24*.

[25] Y. Wang, Z. Ju, X. Tan, L. He, Z. Wu, J. Bian, and S. Zhao, "Audit: audio editing by following instructions with latent diffusion models," in *NeurIPS*, 2023.

[26] M. Elminshawi, S. R. Chetupalli, and E. A. P. Habets, "Dynamic slimmable network for speech separation," *IEEE Signal Processing Letters*, vol. 31, pp. 2205–2209, 2024.

[27] M. Elminshawi, S. R. Chetupalli, and E. Habets, "Slim-tasnet: A slimmable neural network for speech separation," in *WASPAA*, 2023.

[28] S. Chen, Y. Wu, Z. Chen, T. Yoshioka, S. Liu, J. Li, and X. Yu, "Don't shoot butterfly with rifles: Multi-channel continuous speech separation with early exit transformer," in *ICASSP*, 2021.

[29] D. Bralios, E. Tzinis, G. Wichern, P. Smaragdis, and J. L. Roux, "Latent iterative refinement for modular source separation," in *ICASSP*, 2023.

[30] C. Subakan, M. Ravanelli, S. Cornell, M. Bronzi, and J. Zhong, "Attention is all you need in speech separation," 2021. [Online]. Available: https://arxiv.org/abs/2010.13154

[31] E. Perez, F. Strub, H. de Vries, V. Dumoulin, and A. C. Courville, "Film: Visual reasoning with a general conditioning layer," *CoRR*, vol. abs/1709.07871, 2017. [Online]. Available: http://arxiv.org/abs/1709.07871

[32] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210.

[33] G. Wichern, J. Antognini, M. Flynn, L. R. Zhu, E. McQuinn, D. Crow, E. Manilow, and J. L. Roux, "Wham!: Extending speech separation to noisy environments," 2019. [Online]. Available: https://arxiv.org/abs/1907.01160

[34] J. L. Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "Sdr - half-baked or well done?" 2018. [Online]. Available: https://arxiv.org/abs/1811.02508

[35] A. Rix, J. Beerends, M. Hollier, and A. Hekstra, "Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs," in *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*, vol. 2, 2001, pp. 749–752 vol.2.

[36] C. K. A. Reddy, V. Gopal, and R. Cutler, "Dnsmos p.835: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors," 2022. [Online]. Available: https://arxiv.org/abs/2110.01763

[37] C. Veaux, J. Yamagishi, and K. MacDonald, "Cstr vctk corpus: English multi-speaker corpus for speech synthesis," *University of Edinburgh. The Centre for Speech Technology Research*, 2017. [Online]. Available: https://datashare.ed.ac.uk/handle/10283/2651