SYNTHESIZING SPEECH FROM INTRACRANIAL RECORDINGS WITH CONDITIONAL FLOW MATCHING

Anonymous authors
Paper under double-blind review

ABSTRACT

The task of synthesizing speech from neurological data has advanced significantly over the last five years, through a combination of higher-quality data and modern machine-learning techniques. To date, all such approaches have been based on discriminative modeling, i.e. building a deterministic map from neural data to speech audio (e.g., the spectrogram). However, there is reason to believe such models are unnecessarily restrictive. Inspired by the recent evolution in text-to-speech from discriminative to generative models, we reinterpret speech synthesis as a conditional-generation problem. We use conditional flow matching to learn a velocity field that transports a standard normal distribution to the distribution of log-mel spectrogram features, with velocity predictions conditioned on the corresponding neural data by way of a convolutional encoder. Discriminative modeling emerges as the special case in which the model is trained to predict only the initial velocity. We demonstrate our approach on (1) a set of stereo-EEG/speech audio recordings recently made by our group from two participants, and (2) a public data set of microelectrode recordings (single units) made from a person with severe dysarthria. In both cases, our conditional flow outperforms all competitors, in the latter case bringing word error rates of (transcribed) synthesized speech down from about 56% to 22%. Our model also requires about one fourth as many training trials to achieve comparable intelligibility. Overall, we show that conditional flow matching is a powerful approach to neural speech synthesis for datasets ranging from small (sEEG, 0.5 hours) to large (single units, 38 hours).

1 INTRODUCTION

In 2009, Brumberg et al. (2009) demonstrated that vowel formants can be synthesized from the firing rates of a few neurons in the speech-motor cortex. Over the intervening decade and a half, driven by better neural interfaces and modern (artificial) neural networks, speech decoding has progressed to intelligible or nearly intelligible sentences. Still, the lowest word error rates of synthesized speech that has been (automatically) transcribed to text are around 50%, and this with weeks of data acquired from the most invasive microelectrode arrays (Wairagkar et al., 2025), so there is room for improvement.

The problem of speech decoding is naturally formulated in terms of *discriminative modeling*—construct a deterministic map from intracranial neural activity to speech—and accordingly all studies heretofore have adopted this formulation, explicitly or implicitly. The map typically takes the form of a standard neural network for sequence data: some combination of convolutional, recurrent, and attentional (transformer) components. The input data range from aggregate neural activity ($10^4 - 10^5$ neurons) acquired with macroelectrodes like stereo-electroencephalography (sEEG) (Akbari et al., 2019; Verwoert et al., 2022; He et al., 2025) and electrocorticography (ECoG) (Anumanchipalli et al., 2019; Makin et al., 2020; Chen et al., 2024b); to, more recently, spiking activity of single neurons recorded with microelectrode arrays (Willett et al., 2023; Card et al., 2024; Wairagkar et al., 2025). The output could be text or speech audio; in this study, we focus on the latter, because it retains paralinguistic features like intonation, prosody, and volume. In practice, the model is trained to produce not the waveform directly, which has much higher frequency components than the neural data, but intermediate audio features, such as the log-mel spectrogram (Angrick et al.,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

2019; Verwoert et al., 2022), the Bark cepstrum, (Wairagkar et al., 2025), discrete HuBERT speech units (Metzger et al., 2023), handcrafted speech features (Chen et al., 2024b), or the like. These are subsequently transformed into the waveform with an off-the-shelf network like HiFi-GAN (Kong et al., 2020). The imposed loss function is typically mean square error (MSE) in audio-feature space, i.e. Gaussian cross entropy.

The overall approach is sound, but there are two related reasons for thinking it overly restrictive. First, the optimal solution to the MSE penalty is the conditional mean. However, audio features, like other natural signals, are believed to be confined to a curved manifold whose intrinsic dimension is much lower than the ambient space—e.g. the 80-dimensional log-mel space. If conditional probability is distributed over a region of sufficiently high curvature, the conditional mean will not fall on the manifold itself. And we do expect broad conditional distributions, since typical intracranial data underspecify the corresponding audio. Together, this plausibly explains a common failure mode in algorithms for speech synthesis: blurry spectrograms. Instead of "splitting the difference" with the conditional mean, it would be preferable to map each input to a single representative example. This task is closely related to the problem of optimal transport. This suggests replacing the discriminative framework with a conditional generative framework, in which a random initialization determines which representative example will be produced.

In this study, we adapt the conditional flow-matching framework Liu et al. (2023); Lipman et al. (2023); Tong et al. (2024) to the problem of synthesizing speech audio from intracranial data. The neural network is tasked with predicting, not the spectral features themselves (as in the discriminative model), but rather the "velocity" vector that points towards them from the current location (also a network input). At test time, these vectors can be integrated numerically, starting from a random location, into the speech features. In fact, discriminative modeling is the special case of this framework in which the network is trained to solve the problem in a single pass (Appendix A.1). This brings us to the second reason for thinking the discriminative objective is overly restrictive: it requires the model to solve the problem in one step, whereas flow matching allows for iterative refinement over an arbitrary number of steps.

We test this approach on about 0.5 hours of sEEG recordings from two participants and about 38 hours of single-unit recordings from a participant with severe dysarthria. Our results show three main findings. First, conditional flow-matching improves all measures of speech quality, intelligibility, and speaker similarity over published discriminative models on both datasets. Second, on the single-unit dataset, our model achieves a median trial-level word error rate of 0.22, compared with 0.56 for the audio released by Wairagkar et al. (2025). Third, it reaches comparable intelligibility with about one fourth as many training trials. Together, these findings suggest that iterative generation can improve neural speech synthesis over single-step prediction.

2 Related Work

2.1 Speech Synthesis from Intracranial Neural Data

Existing neural speech synthesis methods decouple neural activity decoding from waveform generation: neural signals are first mapped to an intermediate speech representation, which is subsequently converted into audible speech using a pretrained neural vocoder or speech synthesizer, such as WaveNet, LPCNet, or HiFi-GAN. Both handcrafted acoustic/articulatory features and learned speech representations (through self-supervised learning from large-scale speech corpora) have been explored as intermediate targets. Anumanchipalli et al. (2019) used a bidirectional LSTM to predict articulatory kinematic features directly from ECoG, while Angrick et al. (2019) used densely connected convolutional networks to generate mel spectrograms from ECoG. More recently, Metzger et al. (2023) predicted discrete HuBERT units from high-density ECoG using bidirectional recurrent networks and subsequently employed pretrained modules to transform these units into audible speech. Similarly, Li et al. (2024) used Wav2Vec 2.0 speech features as targets and developed a two-stage pipeline for decoding speech from ECoG. Transformer encoder-based architectures have also been explored, with Shigemi et al. (2023) employing a transformer encoder with a parallel WaveGAN vocoder to predict speech from ECoG. Later, Chen et al. (2024b) incorporated 3D Swin Transformer blocks to predict handcrafted speech features directly from a larger ECoG dataset.

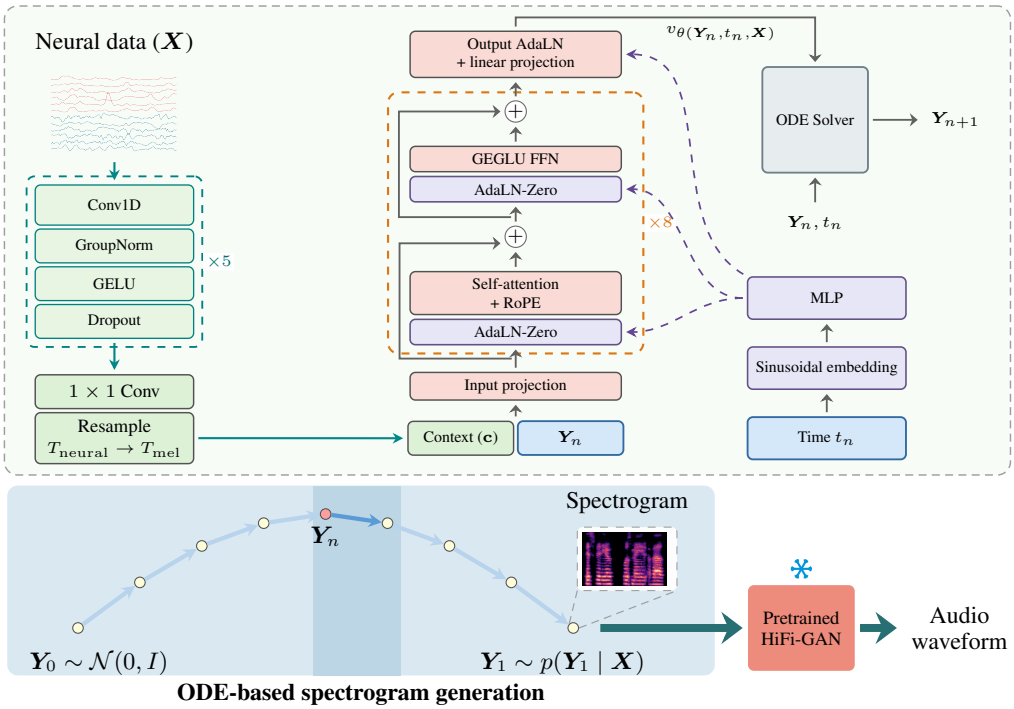


Figure 1: **Conditional flow-matching speech-synthesis pipeline.** The lower panel shows Euler integration transporting Gaussian noise to a log-mel spectrogram, which a pretrained HiFi-GAN (Kong et al., 2020) converts into a speech waveform. The upper panel expands the highlighted Euler step: a neural encoder produces frame-aligned context $\mathbf{c}$ from the neural data, which is concatenated with the current spectrogram.

Apart from ECoG, stereo-electroencephalography (sEEG) has also been explored for speech synthesis. He et al. (2025) used bidirectional GRUs to generate mel spectrograms for Mandarin speech, while Zhang et al. (2025) investigated graph neural networks for speech synthesis from sEEG recorded during speech perception. Lately, microelectrode arrays (Utah arrays) have enabled speech decoding from single-unit neural activity. Wairagkar et al. (2025) demonstrated intelligible speech decoding from a dysarthric participant using session-specific layers for recording drift and a transformer to predict LPCNet-compatible acoustic features.

2.2 Generative Modeling and Flow Matching

Generative modeling has advanced considerably in recent years. In particular, diffusion and score-based models have achieved strong results for modeling high-dimensional data (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song & Ermon, 2019; Song et al., 2021; Dhariwal & Nichol, 2021). Other prominent generative approaches include variational autoencoders, generative adversarial networks, and normalizing flows (Kingma & Welling, 2014; Goodfellow et al., 2014; Rezende & Mohamed, 2015; Kingma & Dhariwal, 2018). Continuous normalizing flows generalize normalizing flows by defining a continuous transformation through an ordinary differential equation (Chen et al., 2018). Recent simulation-free training methods have made these models easier to train (Lipman et al., 2023; Liu et al., 2023). Minibatch optimal-transport coupling can further produce straighter transport paths, reducing the number of function evaluations required during sampling (Tong et al., 2024). Generative models have also been used for text-to-speech synthesis. Diffusion, normalizing-flow, and flow- matching models can generate mel spectrograms conditioned on text (Popov et al., 2021; Kim et al., 2023; 2020; Prenger et al., 2019; Mehta et al., 2024; Le et al., 2023; Eskimez et al., 2024; Chen et al., 2025). In contrast, neural speech synthesis from brain recordings has so far relied primarily on discriminative models that directly regress from neural activity to acoustic representations. This motivates our use of conditional flow matching to model the distribution of speech spectrograms conditioned on neural recordings.

3 METHODS

3.1 PRELIMINARY: FLOW MATCHING

Flow matching (FM) (Lipman et al., 2023; Liu et al., 2023; Tong et al., 2024) is a simulation-free method for training continuous normalizing flows (CNFs) (Chen et al., 2018). A CNF transports a source distribution p_0 to a target distribution p_1 by evolving a sample according to the ordinary differential equation $\mathrm{d}\boldsymbol{y}/\mathrm{d}t = \boldsymbol{u}(\boldsymbol{y}, t)$, where $\boldsymbol{u}(\cdot,\cdot) : \mathbb{R}^d \times [0,1] \to \mathbb{R}^d$ is a time-dependent *velocity field*. This ODE induces a probability path, $p_t(\boldsymbol{y}) \coloneqq \Pr[\boldsymbol{Y}_t = \boldsymbol{y}|t]$. For generative modeling, we choose an easily sampled source distribution, $p_0(\boldsymbol{y}) \triangleq \mathcal{N}(\mathbf{0},\ \mathbf{I})$, and set the target distribution to the data distribution, $p_1(\boldsymbol{y}) \triangleq p_{\mathrm{data}}(\boldsymbol{y})$. Conditional flow matching (CFM) (Lipman et al., 2023; Tong et al., 2024) provides a tractable means of estimating the velocity field, namely by training a neural network $\boldsymbol{v}_{\boldsymbol{\theta}}(\cdot,\cdot)$ to predict the *conditional* velocity from the current "position" $\boldsymbol{y}_t$ and time t, at all positions, time points, and conditions:

$$\mathcal{L}_{\mathrm{CFM}}(\boldsymbol{\theta}) \coloneqq \mathbb{E}_{\boldsymbol{Y}_t, t, \boldsymbol{Z}}\Big[\big\|\boldsymbol{v}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t, t) - \boldsymbol{u}(\boldsymbol{Y}_t, t|\boldsymbol{Z})\big\|^2\Big]. \tag{1}$$

In general, the condition $\boldsymbol{Z}$ is a pair consisting of a source point $\boldsymbol{Y}_0$ and a target point $\boldsymbol{Y}_1$, i.e., $\boldsymbol{Z} \coloneqq (\boldsymbol{Y}_0, \boldsymbol{Y}_1)$. There are different possible choices for the conditional probability paths, but linear paths have been shown to improve the quality of generated samples while requiring fewer neural function evaluations (NFEs). A very simple but useful instance (Tong et al., 2024) is

$$\boldsymbol{Y}_t = (1-t)\boldsymbol{Y}_0 + t\boldsymbol{Y}_1. \tag{2}$$

The corresponding conditional velocity field is $\boldsymbol{Y}_1 - \boldsymbol{Y}_0$.

3.2 SPEECH-SYNTHESIS PIPELINE

We formulate speech synthesis as a *conditional* generation problem.[1] That is, we use a CNF to generate log-mel spectrograms $\boldsymbol{Y}_1$ conditioned on neural recordings $\boldsymbol{X}$. We then convert each generated spectrogram into an audio waveform with a pretrained HiFi-GAN (Kong et al., 2020); see Fig. 1.

3.2.1 CONDITIONAL VELOCITY PREDICTOR

The conditional velocity predictor, $\boldsymbol{v}_{\boldsymbol{\theta}}(\boldsymbol{y}, t, \boldsymbol{x})$, consists of two components: a neural encoder and a diffusion transformer (DiT) (Peebles & Xie, 2023). It takes three inputs: the "interpolated" log-mels ($\boldsymbol{y}_t$), interpolation time (t), and neural recordings ($\boldsymbol{x}$). The neural encoder extracts a conditioning representation from $\boldsymbol{x}$, and the DiT uses this representation together with $\boldsymbol{y}_t$ and t to predict the conditional velocity field. For architectural details, see Fig. 1 and Section A.2.1.

Neural encoder. Neural data are preprocessed before training, with randomly sampled segments paired with their corresponding audio targets. For a segment of length L_{seg} and neural sampling rate f_{neural}, the neural encoder input is $\boldsymbol{X} \in \mathbb{R}^{C \times T_{\mathrm{neural}}}$, where C is the number of retained channels and $T_{\mathrm{neural}} = L_{\mathrm{seg}} f_{\mathrm{neural}}$. The input passes through five convolutional blocks, each consisting of a dilated 1D convolution followed by GroupNorm, GELU, and dropout. The first convolution maps C channels to 256 dimensions, while the remaining blocks operate at 256 dimensions with dilation factors increasing from 1 to 16, expanding the receptive field for richer neural representations. A 1×1 convolution then projects the features to 192 dimensions, which are resampled and aligned to the corresponding spectrogram frames, followed by LayerNorm. The resulting $\mathbf{c} \in \mathbb{R}^{T_{\mathrm{mel}} \times 192}$ is provided as an additional condition to the DiT-based conditional velocity predictor.

Diffusion transformer. An input projection maps this to a 384-dimensional hidden representation, which is processed by eight DiT blocks, each comprising six-head multi-head self-attention and a GEGLU feed-forward network, with RoPE providing positional information. The resulting features pass through an output AdaLN layer and linear projection to produce the conditional velocity prediction, which has the same shape as the input spectrogram and represents its instantaneous velocity. The time condition is encoded using a sinusoidal embedding followed by an MLP, whose

[1] This conditioning, on intracranial data, is not to be confused with the conditioning on source and target points that gives "conditional flow matching" its name.

output modulates the AdaLN-Zero layers within the DiT blocks and the output AdaLN layer while gating the residual attention and feed-forward branches.

3.2.2 TRAINING OBJECTIVE

We train the conditional velocity predictor using the conditional flow-matching objective. Conceptually, we can think of conditional spectrogram generation as transporting the source distribution $\boldsymbol{Y}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ to the target distribution $\boldsymbol{Y}_1 \sim p(\boldsymbol{y}_1|\boldsymbol{x})$. For each training example, we sample a target spectrogram $\boldsymbol{Y}_1$ and its paired neural recording $\boldsymbol{X}$, a noise vector $\boldsymbol{Y}_0 \sim \mathcal{N}(0, I)$, and a time $t \sim U[0, 1]$. Noise vectors and target spectrograms within each minibatch are paired using minibatch optimal transport. In the language of conditional flow matching (Lipman et al., 2023), then, the condition variable is $\boldsymbol{Z} \coloneqq (\boldsymbol{Y}_0, \boldsymbol{Y}_1, \boldsymbol{X})$, distributed as $\pi(\boldsymbol{Y}_0, \boldsymbol{Y}_1)p_{\text{data}}(\boldsymbol{X} \mid \boldsymbol{Y}_1)$. Here, $\pi(\boldsymbol{Y}_0, \boldsymbol{Y}_1)$ couples only the noise source and target spectrograms; the observed pairing between $\boldsymbol{Y}_1$ and $\boldsymbol{X}$ is preserved. We use the linear path given by Eq. 2 and its corresponding conditional velocity as target. In this case, the CFM loss (Eq. 1) becomes

$$\mathcal{L}_{\text{CFM}}(\boldsymbol{\theta}) \coloneqq \mathbb{E}_{t,\boldsymbol{Z}}\Big[\|\boldsymbol{v}_{\boldsymbol{\theta}}(\boldsymbol{Y}_t, t, \boldsymbol{X}) - (\boldsymbol{Y}_1 - \boldsymbol{Y}_0)\|^2\Big]. \tag{3}$$

Note that the neural recording does not define the source sample; instead, it conditions the velocity field that transports Gaussian noise toward the corresponding speech spectrogram.

3.2.3 SAMPLING

Once trained, the conditional velocity predictor approximates the dynamics of the CNF. To generate a spectrogram, we draw an initial sample $\boldsymbol{y}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and numerically integrate the ODE $\mathrm{d}\boldsymbol{y}_t/\mathrm{d}t = \boldsymbol{v}_{\boldsymbol{\theta}}(\boldsymbol{y}_t, t, \boldsymbol{x})$, from $t = 0$ to $t = 1$, conditioned on the neural recording $\boldsymbol{x}$. We use the Euler method,

$$\boldsymbol{y}_{n+1} = \boldsymbol{y}_n + \Delta_t \boldsymbol{v}_{\boldsymbol{\theta}}(\boldsymbol{y}_n, t_n, \boldsymbol{x}), \tag{4}$$

where $t_n = n\Delta_t$, $\Delta_t = 1/N$, and N is the number of neural function evaluations. The final state $\boldsymbol{y}_N$ is the generated spectrogram, which we convert to an audio waveform using a pretrained HiFi-GAN (Kong et al., 2020). Except where otherwise stated, we use $N = 8$ steps in all experiments. To reduce bias, we compute each metric for $K = 8$ samples per trial, and report average values of the metric. The full training/testing details are described in Section A.2.

4 EXPERIMENTS

4.1 DATASETS

4.1.1 NZ DATASETS

Our in-house NZ datasets consist of sEEG recordings from two epilepsy patients (one male, NZ0; one female, NZ1) undergoing seizure focus localization via surgically implanted depth electrodes. The datasets each consist of approximately 30 minutes of sEEG and paired audio recordings per participant, collected during speech production trials from an English corpus. Participant NZ0 was implanted with 17 depth electrodes (8–18 contacts per electrode), and NZ1 with 13 depth electrodes (8–15 contacts per electrode), yielding approximately 200 channels per participant. Electrode placement was clinically determined based on putative seizure loci. All electrophysiological signals were recorded using a Natus Quantum amplifier at a sampling rate of 2048 Hz. Intracranial data were pre-processed according to a standard protocol: Raw channels were notch-filtered at the line frequency, common-average referenced within each electrode, and downsampled to 200 Hz. Activity in the high-γ band was passed through a bank of eleven constant-Q, Gaussian filters, with center frequencies log-spaced between 70 and 200 Hz. The analytic amplitude was computed in each subband and averaged together. Local field potentials (LFPs) were isolated by low-pass filtering below 50 Hz. To eliminate irrelevant channels, a gradient-based saliency analysis was applied to a crude discriminative decoder trained on a subset of the data. In theory this implies some information leak into the results, but in practice the leak is tiny, and in any case the same set of electrodes was used for all models, competitors as well as our own. We retained 32 high-γ channels and 32 LFP channels per participant. Reference speech waveforms were converted into 80-bin log-mel spectrograms using the HiFi-GAN preprocessing pipeline (Kong et al., 2020). A comprehensive description of the experimental protocol and complete preprocessing chain is provided in Appendix A.3.

4.1.2 T15 DATASET

We also used the dataset released by Wairagkar et al. (2025), recorded from participant T15, an adult male with severe dysarthria, implanted with four 64-electrode Utah arrays in the left ventral precentral gyrus. The dataset comprises 37 sessions over post-implant days 25–195 and 8,489 trials (38.1 h), in each of which the participant attempted to speak a unique cued sentence. Neural signals were processed online into spike-band power and threshold-crossing counts ("spikes") per electrode, binned at 10 ms and further processed, yielding a 512-dimensional feature series at 100 Hz, which we use exactly as released without further normalization. Because T15 could not produce intelligible speech, Wairagkar et al. (2025) synthesized acoustic targets by rendering each cued sentence with a text-to-speech voice, encoding it into 20-dimensional LPCNet vocoder features (Valin & Skoglund, 2019) per 10 ms frame, and finally aligning it to the neural data using the participant's attempted-speech timings. We use the syllable-aligned targets, as the authors did for all sessions after the first. Eight sessions (days 172–195) were designated evaluation sessions in the original study, and the public release includes the audio synthesized online by the authors' decoder for every evaluation trial, which we adopt as our benchmark. By training on the same neural features and speech targets as Wairagkar et al., our results are directly comparable to theirs. Full details are given in the original study (Wairagkar et al., 2025).

4.2 SYNTHESIS FROM SPEAKING PARTICIPANTS (STEREO-EEG)

In this section, we present the main synthesis results on our in-house sEEG datasets, participants NZ0 and NZ1. We implement five benchmark models, four of which are end-to-end models, while the algorithm of Metzger et al. (2023) uses a modular pipeline. We also compare our method with an architecture-matched CNN-DiT model trained to predict mel spectrograms directly using MSE loss. All benchmarks were extensively tuned on the NZ datasets for hyperparameter selection prior to comparison. Performance was evaluated using two complementary categories of metrics: spectral fidelity and perceptual quality. Spectral fidelity was assessed using the Pearson correlation coefficient evaluated on the speech frames (r_{sp}), mel-cepstral distortion (MCD), and extended short-time objective intelligibility (ESTOI). Perceptual quality and speaker-identity preservation were evaluated using UTMOS (Saeki et al., 2022), a reference-free mean opinion score (MOS) predictor; and speaker similarity (SpkSim), derived from *ECAPA-TDNN* (Desplanques et al., 2020). The final step of the algorithm of Metzger et al. (2023) is voice cloning, which achieves arbitrarily high UTMOS scores—indeed, much higher than our "ceiling," derived by computing the spectrogram from the raw audio and then reconstituting it with HiFi-GAN—and even when the predictions are completely unintelligible (ESTOI < 0.1). We therefore excluded it from the UTMOS-based rankings.

Table 1 compares our model with five benchmark methods and the architecture-matched CNN-DiT mel regressor trained with MSE loss across all evaluation metrics. We first observe that our proposed flow-based method achieves the highest performance across both fidelity and perceptual metrics. We also note that the effect sizes are not similar across all metrics. Most notably in the perceptual axes, our method achieves the highest performance boosts (e.g., 0.649 vs. 0.520 achieved by CNN-LSTM on SpkSim and 1.882 vs. 1.490 achieved by Wairagkar et al. (2025) on UTMOS in NZ0) and gets UTMOS scores very close to the vocoded ground truth limit (e.g., in NZ1, an UTMOS score of 2.022 vs. the ground truth ceiling of 2.025). On average we achieve $\sim$25% gains in both SpkSim and UTMOS over the next best benchmark method across both participants. We also observe that gains in the fidelity axis are modest in comparison to the perceptual axis, except the intelligibility metric ESTOI. The correlation coefficient has the smallest effect size ($\sim$1.0% gains on average across both NZ0 and NZ1), followed by MCD ($\sim$4.4%) and ESTOI ($\sim$16%). This is consistent with our central hypothesis (Section 1): Discriminative models can produce low MSE (or, equivalently, high correlations r_{sp}), but at the price of spectral smearing and therefore lower perceptual quality. Our architecture-matched MSE regressor follows this pattern: it achieves good r_{sp} and MCD scores, but lower intelligibility. The proposed generative model improves ESTOI, speaker similarity, and UTMOS while maintaining comparable r_{sp} and MCD.

4.3 SYNTHESIS FROM A SEVERELY DYSARTHRIC PARTICIPANT (MICROELECTRODE ARRAY)

We turn to the data collected by (Wairagkar et al., 2025) from a severely dysarthric person. In comparison with the NZ datasets, these data are both far more spatially resolved (single units vs.

Table 1: **Synthesis results on the NZ datasets under leave-one-block-out cross-validation (NZ0: 8 folds; NZ1: 9 folds).** Mean ± s.d. are reported across folds. r_{sp}, ESTOI and MCD are computed under dynamic time warping on speech frames only. Best in **bold**, second best underlined. Markers on our row are a one-sided Wilcoxon signed-rank test paired over folds against the strongest published baseline on that metric ($^{***}\,p < 0.005$, $^{**}\,p < 0.01$, $^{*}\,p < 0.05$, † tested, not significant). *Metzger et al. (2023) attains a high reference-free UTMOS on audio that is not intelligible (ESTOI < 0.1), so this entry is excluded from the rankings (see Section 4.2).The DiT mel regressor is our architecture-matched, MSE-trained control.

	Model	r_{sp} ↑	SpkSim ↑	ESTOI ↑	UTMOS ↑	MCD ↓
NZ0	Transformer (Shigemi et al., 2023)	0.686 ±0.039	0.342 ±0.076	0.184 ±0.034	1.225 ±0.059	7.003 ±0.825
	CNN-LSTM (Quadri et al., 2024)	0.767 ±0.015	0.520 ±0.048	0.280 ±0.032	1.438 ±0.076	7.074 ±0.684
	Metzger et al. (2023)	0.586 ±0.025	0.185 ±0.017	0.091 ±0.025	3.016* ±0.104	8.592 ±0.803
	VocalMind (He et al., 2025)	0.793 ±0.013	0.462 ±0.100	0.319 ±0.058	1.361 ±0.056	7.194 ±0.865
	Wairagkar et al. (2025)	0.800 ±0.015	0.346 ±0.058	0.298 ±0.036	1.490 ±0.127	6.862 ±0.754
	CNN-DiT mel regressor	**0.804** ±0.014	0.540 ±0.076	0.324 ±0.036	1.577 ±0.112	6.767 ±0.875
	Ours	0.803† ±0.013	**0.649***** ±0.052	**0.373***** ±0.035	**1.882***** ±0.080	**6.526*** ±0.618
	HiFi-GAN vocoded GT	0.966 ±0.005	0.952 ±0.005	0.861 ±0.017	2.043 ±0.149	2.416 ±0.323
NZ1	Transformer (Shigemi et al., 2023)	0.661 ±0.040	0.356 ±0.050	0.202 ±0.039	1.382 ±0.120	6.073 ±0.375
	CNN-LSTM (Quadri et al., 2024)	0.708 ±0.039	0.514 ±0.045	0.295 ±0.037	1.512 ±0.068	5.480 ±0.516
	Metzger et al. (2023)	0.483 ±0.047	0.227 ±0.031	0.081 ±0.019	3.001* ±0.063	7.654 ±0.318
	VocalMind (He et al., 2025)	0.739 ±0.025	0.507 ±0.053	0.342 ±0.046	1.515 ±0.153	5.326 ±0.406
	Wairagkar et al. (2025)	0.750 ±0.039	0.387 ±0.040	0.321 ±0.040	1.565 ±0.063	5.360 ±0.427
	CNN-DiT mel regressor	0.759 ±0.034	0.622 ±0.023	0.324 ±0.038	1.648 ±0.188	5.283 ±0.493
	Ours	**0.762*** ±0.031	**0.653***** ±0.030	**0.393***** ±0.044	**2.022***** ±0.168	**5.117**† ±0.511
	HiFi-GAN vocoded GT	0.951 ±0.012	0.929 ±0.009	0.837 ±0.059	2.025 ±0.151	2.140 ±0.143

aggregate activity of $\sim 10^4$ neurons) and far more numerous ($\sim$38 vs. 0.5 hours). Consequently, it is possible to synthesize intelligible speech, and word error rate (WER) of transcribed speech becomes a useful metric. Here we report performance under the metrics used by Wairagkar et al. (2025): median (across trials) WER, correlation coefficient (with dynamic time warping alignment), MCD, and additionally the ESTOI. All other speech metrics are reported in Section A.4.5. For reproducibility of WER, we replace the human transcribers of the original study with automatic speech recognition under Whisper (Radford et al., 2023). As targets for this dysarthric participant we used the proxy speech synthesized and aligned by Wairagkar et al. (2025) (see Section 4.1.2).

We ran two experiments. First, we trained our flow model on the full dataset following the original training protocol and evaluated each trial from the official test days (172–195), using the LPCNet vocoder employed by Wairagkar et al. (2025) rather than HiFi-GAN. Following the Wairagkar et al., we used strictly causal variants of our flow model and matched MSE-regressor (matched receptive field to 610 ms, Wairagkar et al. uses 600 ms) for training/evaluating on this dataset. Both our models outperform Wairagkar et al. (see Tables 2 and 12), reducing the median WER by $\sim$60%, down to 22%—the outer bound of acceptable intelligibility (Munteanu et al., 2006). On the high-quality T15 dataset, word intelligibility is seemingly set by the architecture and the causal protocol. The flow objective enhances the quality of the synthesized speech—ESTOI (DTW) rises from 0.355 to 0.404 ($+14\%$), similar gains are also seen across UTMOS, SPkSim and MCD (see Table 12).

Second, we fixed evaluation to day 172 and retrained both our flow model and Wairagkar et al. (2025) from scratch on the N most recent prior sessions ($N \in \{1, 2, 4, 8, 16, 29\}$, identical splits, both evaluated offline). Ours is more intelligible at every budget, and with four sessions (859 trials) already matches the best WER the baseline attains at any budget (0.50 vs. 0.53 at 16 sessions)—a four-fold reduction in recording sessions (see Fig. 2), and evidence that flow-based synthesis can substantially reduce the neural data a practical speech prosthesis would require.

4.4 Ablations and Additional Analyses

4.4.1 Effect of ODE-Solver Budget and the Distortion–Perception Trade-Off

The number N of Euler steps (or equivalently, NFEs) taken at test time is not fixed by training and can be chosen freely. Heretofore, we have used $N = 8$. In Fig. 3 we show the effect of N on perceptual quality (vertical axis) and fidelity (horizontal axis). As N increases, perceptual

Table 2: **Speech synthesis from T15.** Following Wairagkar et al. (2025), we report mean ± s.d. across 964 trials for spectral correlation (r_{sp}), MCD (dB) with silent periods, and ESTOI; and median and interquartile range for WER. r_{sp}, MCD and ESTOI are all computed after DTW alignment, as in the original study. Stars indicate significance under a signed-rank test with the convention of Table 1.

Method	WER ↓	r_{sp} ↑	MCD w/ sil. ↓	ESTOI ↑
Wairagkar et al.	0.56 [0.29–0.83]	0.850 ± 0.039	2.96 ± 0.53	0.25 ± 0.06
CNN-DiT regressor	0.22 [0.00–0.44]	0.890 ± 0.036	2.43 ± 0.45	0.35 ± 0.06
Ours	**0.22 [0.00–0.40]*****	**0.898 ± 0.037*****	**2.37 ± 0.45*****	**0.40 ± 0.07*****

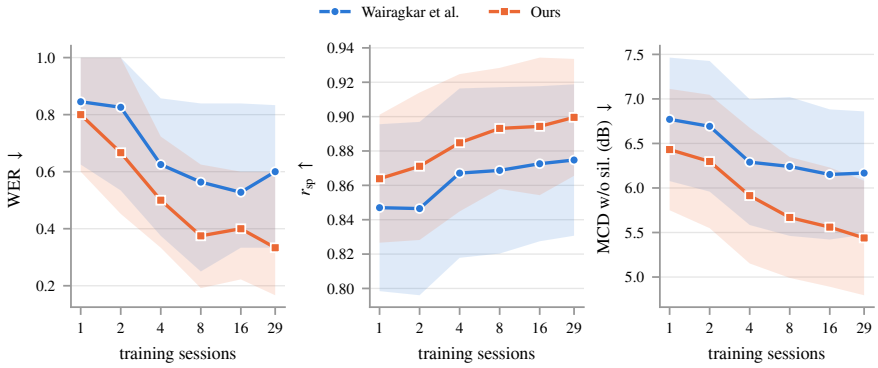


Figure 2: **Data efficiency on T15 (Wairagkar et al., 2025).** Both decoders are trained from scratch on the N most recent sessions before day 172 and evaluated on its 112 held-out trials. Points: per-trial WER (median), r_{sp}, and MCD (mean); bands: IQR and ± s.d. over trials.

quality improves, but fidelity, measured here by r_{sp}, modestly decreases. This is consistent with our hypothesis that the mean-seeking, coarse estimates produced by a small N—or, in the limiting case, a discriminative model (see Section A.1)—can achieve superior spectral correlations by "splitting the difference" between plausible solutions. But the price is a decrease in perceptual quality, since the mean need not resemble realistic spectral features. This gives rise to a distortion–perception trade-off, as observed in image processing (Blau & Michaeli, 2017).

4.4.2 Neural Signal Conditioning and the Source Distribution

It is essential that the intracranial data influence the generation of spectral features in the flow model, but there is more than one way to do this. In the model discussed up to this point, each trajectory is initialized at a draw from a standard normal distribution, and the velocity-predicting network is provided the intracranial data as input. Alternatively, we might initialize the trajectory near a point determined by the neural data and do without conditioning. Table 3 (first and second rows, resp.) shows performance under these two variations on conditional flow matching. Performance is similar, although conditioning (first row) is generally superior. Cutting the convolution encoder network down to a single linear layer (third row) degrades performance further still. Initializing the trajectory at precisely the encoded neural data, i.e. without adding Gaussian noise, devastates performance, especially perceptual metrics. This is again consistent with our hypothesis (see Section 1) that random initialization discourages mean-seeking behavior.

5 Conclusion

Decoding speech, and particularly synthesizing speech, from intracranial recordings is on the cusp of clinical and commercial viability. The present study takes the field substantially closer to these goals, generating intelligible speech from microelectrode recordings made from a severely dysarthric person attempting to speak: Median word error rates of transcriptions of the synthesized speech are just over 20%; WERs under the previous state of the art were above 50%. Our approach likewise improves (over existing approaches) the perceptual quality of speech decoded from about half

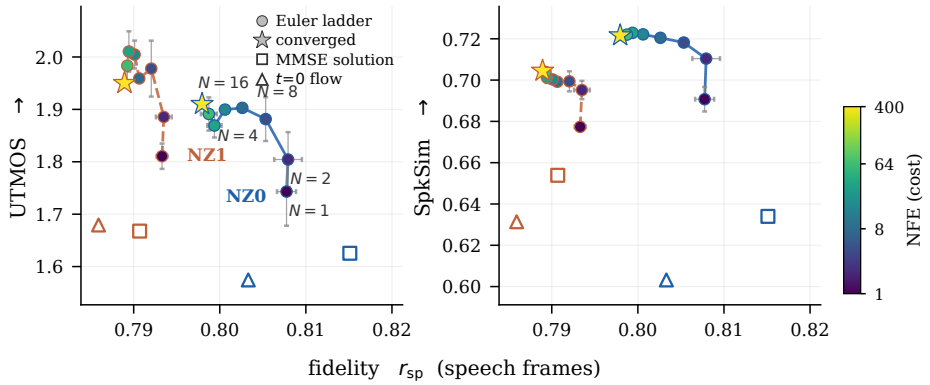


Figure 3: **Effect of the number of Euler steps on fidelity and perceptual quality.** Fidelity is on the horizontal axis and perceptual quality on the vertical axis. Circles show Euler sampling with $N = 1, \ldots, 64$ steps. The star shows the reference solution, using $N = 400$. Color shows N/NFEs; solid lines are NZ0 and dashed lines are NZ1. Open markers show an MSE regressor (□) and a flow model trained only at $t = 0$ and sampled in one step (△). Euler runs with the same seed start from the same x_0. Error bars show the standard deviation across samples generated with three seeds per trial. Both participants use held-out block 4.

Table 3: **Integration of sEEG into the flow model.** Each variant is evaluated on NZ0 block 4 with three separate seeds and the mean ± s.d. is reported. $E(\boldsymbol{x})$ denotes the output of the neural context encoder CNN for any sEEG segment $\boldsymbol{x}$. The first row corresponds to the model used in the rest of this study.

Source	**Cond.**	r_{sp} ↑	SpkSim ↑	ESTOI ↑	UTMOS ↑	MCD ↓
$x_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$	$E(\boldsymbol{x})$	**0.803** ±0.002	**0.717** ±0.008	**0.403** ±0.000	1.903 ±0.005	6.892 ±0.097
$x_0 \sim \mathcal{N}(\mathbf{W}E(\boldsymbol{x}) + \boldsymbol{b}, \sigma^2\mathbf{I})$	–	0.798 ±0.002	0.702 ±0.003	0.387 ±0.004	**1.914** ±0.033	7.066 ±0.070
$x_0 \sim \mathcal{N}(\mathbf{W}\boldsymbol{x} + \boldsymbol{b}, \sigma^2\mathbf{I})$	–	0.794 ±0.006	0.664 ±0.009	0.343 ±0.007	1.778 ±0.016	6.993 ±0.115
$x_0 = \mathbf{W}E(\boldsymbol{x}) + \boldsymbol{b}$	–	0.544 ±0.068	0.056 ±0.035	-0.030 ±0.028	0.817 ±0.052	**6.433** ±0.303
$x_0 = \mathbf{W}\boldsymbol{x} + \boldsymbol{b}$	–	0.730 ±0.025	0.278 ±0.046	0.181 ±0.038	1.068 ±0.070	6.692 ±0.037

an hour of stereo-EEG data—about 25% improvements on SpkSim and UTMOS over the nearest benchmark—although only occasionally to the level of true intelligibility.

Over the last 15 years, neural networks have displaced competing algorithms across vast areas of science and engineering. A well known and discussed cause is their scaling behavior: performance can be increased seemingly arbitrarily with data, simply by making the networks bigger—a property well suited to the era of large datasets made possible by the internet. However, neurological datasets are typically several orders of magnitude smaller than modern ML datasets, so the source of neural-network success in this area must be sought elsewhere. An attractive and fundamental feature of neural networks that does not depend on large data sets can be summarized as *iterative refinement*. The most obvious example is gradient descent, but data are iteratively refined not just in learning but in inference: within a layer in recurrent networks, across layers in all networks (especially conspicuous in fixed-width networks with residual connections, like transformers), and across network evaluations in diffuson and flow-matching models. From this perspective, what distinguishes conditional generative models from discriminative models is not just the appearance of a noise distribution, but the fact that more than one step (NFE) can be taken to "refine" the input into the output.

We showed that this iterative refinement process can improve speech synthesis results by significant margins. There is, however, a downside to this approach, which is that more NFEs make real-time decoding harder. More generally, we have not demonstrated real-time decoding in this study, even when the diffusion transformer and convolutional encoder are made causal (as in our experiments with T15), since this would require the construction of a streaming pipeline. We consider this an important avenue for future work.

AI USE STATEMENT

LLMs (primarily Opus and also Sonnet from the Claude LLM family) were used throughout the project for executing research ideas via coding assistance. No LLMs were used for formulating the research question or designing the methodology.

ETHICS STATEMENT

The sEEG study was approved by the local Institutional Review Board (IRB). Both participants provided written informed consent, and data were collected under medical supervision. All authors who accessed the nonpublic participant data completed the institution's required HIPAA privacy training. We also analyzed the publicly released, de-identified T15 dataset (Wairagkar et al., 2025). Its collection and consent procedures are described in the original study.

REFERENCES

Hassan Akbari, Bahar Khalighinejad, Jose L. Herrero, Ashesh D. Mehta, and Nima Mesgarani. Towards reconstructing intelligible speech from the human auditory cortex. *Scientific Reports*, 9(1), 2019. doi: 10.1038/s41598-018-37359-z. URL https://doi.org/10.1038/s41598-018-37359-z.

Miguel Angrick, Christian Herff, Emily Mugler, Matthew C Tate, Marc W Slutzky, Dean J Krusienski, and Tanja Schultz. Speech synthesis from ECoG using densely connected 3D convolutional neural networks. *Journal of Neural Engineering*, 16(3):036019, 2019. doi: 10.1088/1741-2552/ab0c59. URL https://doi.org/10.1088/1741-2552/ab0c59.

Gopala K. Anumanchipalli, Josh Chartier, and Edward F. Chang. Speech synthesis from neural decoding of spoken sentences. *Nature*, 568(7753):493–498, 2019. doi: 10.1038/s41586-019-1119-1. URL https://doi.org/10.1038/s41586-019-1119-1.

Yochai Blau and Tomer Michaeli. The Perception-Distortion Tradeoff. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6228-6237, 2018*, 2017. doi: 10.1109/CVPR.2018.00652. URL https://arxiv.org/abs/1711.06077.

Jonathan S. Brumberg, Philip R. Kennedy, and Frank H. Guenther. Artificial speech synthesizer control by brain-computer interface. In *Interspeech 2009*. ISCA, 2009. doi: 10.21437/interspeech.2009-225. URL https://doi.org/10.21437/interspeech.2009-225.

Nicholas S. Card, Maitreyee Wairagkar, Carrina Iacobacci, Xianda Hou, Tyler Singer-Clark, Francis R. Willett, Erin M. Kunz, Chaofei Fan, Maryam Vahdati Nia, Darrel R. Deo, Aparna Srinivasan, Eun Young Choi, Matthew F. Glasser, Leigh R. Hochberg, Jaimie M. Henderson, Kiarash Shahlaie, Sergey D. Stavisky, and David M. Brandman. An Accurate and Rapidly Calibrating Speech Neuroprosthesis. *New England Journal of Medicine*, 391(7):609–618, 2024. doi: 10.1056/nejmoa2314132. URL https://doi.org/10.1056/nejmoa2314132.

Defang Chen, Zhenyu Zhou, Jian-Ping Mei, Chunhua Shen, Chun Chen, and Can Wang. A Geometric Perspective on Diffusion Models. *arXiv preprint*, 2024a. URL https://arxiv.org/abs/2305.19947.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural Ordinary Differential Equations. In *Advances in Neural Information Processing Systems*, 2018. URL https://arxiv.org/abs/1806.07366.

Xupeng Chen, Ran Wang, Amirhossein Khalilian-Gourtani, Leyao Yu, Patricia Dugan, Daniel Friedman, Werner Doyle, Orrin Devinsky, Yao Wang, and Adeen Flinker. A neural speech decoding framework leveraging deep learning and speech synthesis. *Nature Machine Intelligence*, 6(4):467–480, 2024b. doi: 10.1038/s42256-024-00824-8. URL https://doi.org/10.1038/s42256-024-00824-8.

Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, Jian Zhao, Kai Yu, and Xie Chen. F5-TTS: A fairytaler that fakes fluent and faithful speech with flow matching. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1:*

Long Papers), pp. 6255–6271. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.313. URL https://aclanthology.org/2025.acl-long.313/.

Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck. ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification. In *Interspeech 2020*. ISCA, 2020. doi: 10.21437/interspeech.2020-2650. URL https://doi.org/10.21437/interspeech.2020-2650.

Prafulla Dhariwal and Alexander Nichol. Diffusion Models Beat GANs on Image Synthesis. In *Advances in Neural Information Processing Systems*, 2021. URL https://arxiv.org/abs/2105.05233.

Sefik Emre Eskimez, Xiaofei Wang, Manthan Thakker, Canrun Li, Chung-Hsien Tsai, Zhen Xiao, Hemin Yang, Zirun Zhu, Min Tang, Xu Tan, Yanqing Liu, Sheng Zhao, and Naoyuki Kanda. E2 TTS: Embarrassingly easy fully non-autoregressive zero-shot TTS, 2024. URL https://arxiv.org/abs/2406.18009.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. In *Advances in Neural Information Processing Systems*, 2014. doi: 10.1145/3422622. URL https://doi.org/10.1145/3422622.

Tianyu He, Mingyi Wei, Ruicong Wang, Renzhi Wang, Shiwei Du, Siqi Cai, Wei Tao, and Haizhou Li. VocalMind: A Stereotactic EEG Dataset for Vocalized, Mimed, and Imagined Speech in Tonal Language. *Scientific Data*, 12(1), 2025. doi: 10.1038/s41597-025-04741-2. URL https://doi.org/10.1038/s41597-025-04741-2.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models. In *Advances in Neural Information Processing Systems*, 2020. URL https://arxiv.org/abs/2006.11239.

Heeseung Kim, Sungwon Kim, and Sungroh Yoon. Guided-TTS: A Diffusion Model for Text-to-Speech via Classifier Guidance. In *ICML 2022*, 2023. URL https://openreview.net/forum?id=W3SxRos66mT.

Jaehyeon Kim, Sungwon Kim, Jungil Kong, and Sungroh Yoon. Glow-TTS: A Generative Flow for Text-to-Speech via Monotonic Alignment Search. In *Advances in Neural Information Processing Systems*, 2020. URL https://arxiv.org/abs/2005.11129.

Diederik P. Kingma and Prafulla Dhariwal. Glow: Generative Flow with Invertible 1x1 Convolutions. In *Advances in Neural Information Processing Systems*, 2018.

Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes. *arXiv preprint*, 2014. URL https://arxiv.org/abs/1312.6114.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis. In *Advances in Neural Information Processing Systems*, 2020.

Matthew Le, Apoorv Vyas, Bowen Shi, Brian Karrer, Leda Sari, Rashel Moritz, Mary Williamson, Vimal Manohar, Yossi Adi, Jay Mahadeokar, and Wei-Ning Hsu. Voicebox: Text-guided multilingual universal speech generation at scale. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://papers.nips.cc/paper_files/paper/2023/hash/2d8911db9ecedf866015091b28946e15-Abstract-Conference.html.

Jiawei Li, Chunxu Guo, Li Fu, Lu Fan, Edward F Chang, and Yuanning Li. Neural2speech: a transfer learning framework for neural-driven speech reconstruction. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2200–2204. IEEE, 2024.

Y. Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow Matching for Generative Modeling. In *International Conference on Learning Representations*, 2023. URL https://www.semanticscholar.org/paper/af68f10ab5078bfc519caae377c90ee6d9c504e9.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow. In *ICLR 2023*, 2023. URL https://arxiv.org/abs/2209.03003.

Joseph G. Makin, David A. Moses, and Edward F. Chang. Machine translation of cortical activity to text with an encoder–decoder framework. *Nature Neuroscience*, 23(4):575–582, 2020. doi: 10.1038/s41593-020-0608-8. URL https://doi.org/10.1038/s41593-020-0608-8.

Shivam Mehta, Ruibo Tu, Jonas Beskow, Éva Székely, and Gustav Eje Henter. Matcha-TTS: A fast TTS architecture with conditional flow matching. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024. URL https://arxiv.org/abs/2309.03199.

Sean L. Metzger, Kaylo T. Littlejohn, Alexander B. Silva, David A. Moses, Margaret P. Seaton, Ran Wang, Maximilian E. Dougherty, Jessie R. Liu, Peter Wu, Michael A. Berger, Inga Zhuravleva, Adelyn Tu-Chan, Karunesh Ganguly, Gopala K. Anumanchipalli, and Edward F. Chang. A high-performance neuroprosthesis for speech decoding and avatar control. *Nature*, 620(7976):1037–1046, 2023. doi: 10.1038/s41586-023-06443-4. URL https://doi.org/10.1038/s41586-023-06443-4.

Cosmin Munteanu, Gerald Penn, Ron Baecker, Elaine Toms, and David James. Measuring the acceptable word error rate of machine-generated webcast transcripts. In *Interspeech 2006*. ISCA, 2006. doi: 10.21437/interspeech.2006-40. URL https://doi.org/10.21437/interspeech.2006-40.

William Peebles and Saining Xie. Scalable Diffusion Models with Transformers. In *IEEE International Conference on Computer Vision*, 2023. doi: 10.1109/ICCV51070.2023.00387. URL https://arxiv.org/abs/2212.09748.

Vadim Popov, Ivan Vovk, Vladimir Gogoryan, Tasnima Sadekova, and Mikhail Kudinov. Grad-TTS: A Diffusion Probabilistic Model for Text-to-Speech. In *Proceedings of the 38th International Conference on Machine Learning (ICML 2021)*, 2021. URL https://arxiv.org/abs/2106.06103.

Ryan Prenger, Rafael Valle, and Bryan Catanzaro. Waveglow: A Flow-based Generative Network for Speech Synthesis. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019. doi: 10.1109/icassp.2019.8683143. URL https://doi.org/10.1109/icassp.2019.8683143.

Zeenat Firdosh Quadri, M. Saqib Akhoon, and Sajad A. Loan. Epileptic Seizure Prediction Using Stacked CNN-BiLSTM: A Novel Approach. *IEEE Transactions on Artificial Intelligence*, 5(11):5553–5560, 2024. doi: 10.1109/tai.2024.3410928. URL https://doi.org/10.1109/tai.2024.3410928.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust Speech Recognition via Large-Scale Weak Supervision. In *Proceedings of the 40th International Conference on Machine Learning (ICML 2023)*, 2023. URL https://arxiv.org/abs/2212.04356.

Danilo Jimenez Rezende and Shakir Mohamed. Variational Inference with Normalizing Flows. In *ICML 2015*, 2015. URL https://openreview.net/forum?id=rkVNToWO-S.

Takaaki Saeki, Detai Xin, Wataru Nakata, Tomoki Koriyama, Shinnosuke Takamichi, and Hiroshi Saruwatari. UTMOS: UTokyo-SaruLab System for VoiceMOS Challenge 2022. In *Interspeech 2022*. ISCA, 2022. doi: 10.21437/interspeech.2022-439. URL https://doi.org/10.21437/interspeech.2022-439.

Kai Shigemi, Shuji Komeiji, Takumi Mitsuhashi, Yasushi Iimura, Hiroharu Suzuki, Hidenori Sugano, Koichi Shinoda, Kohei Yatabe, and Toshihisa Tanaka. Synthesizing Speech from ECoG with a Combination of Transformer-Based Encoder and Neural Vocoder. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023. doi: 10.1109/icassp49357.2023.10097004. URL https://doi.org/10.1109/icassp49357.2023.10097004.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. In *ICLR (Workshop Poster) 2014*, 2014. URL https://arxiv.org/abs/1312.6034.

Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep Unsupervised Learning Using Nonequilibrium Thermodynamics. In *Proceedings of the 32nd International Conference on Machine Learning (ICML 2015)*, 2015. URL https://arxiv.org/abs/1503.03585.

Yang Song and Stefano Ermon. Generative Modeling by Estimating Gradients of the Data Distribution. In *Advances in Neural Information Processing Systems*, 2019. URL https://arxiv.org/abs/1907.05600.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-Based Generative Modeling through Stochastic Differential Equations. In *ICLR 2021*, 2021. URL https://openreview.net/forum?id=PxTIG12RRHS.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *arXiv preprint*, 2024. URL https://arxiv.org/abs/2302.00482.

Jean-Marc Valin and Jan Skoglund. LPCNET: Improving Neural Speech Synthesis through Linear Prediction. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019. doi: 10.1109/icassp.2019.8682804. URL https://doi.org/10.1109/icassp.2019.8682804.

Maxime Verwoert, Maarten C. Ottenhoff, Sophocles Goulis, Albert J. Colon, Louis Wagner, Simon Tousseyn, Johannes P. van Dijk, Pieter L. Kubben, and Christian Herff. Dataset of Speech Production in intracranial Electroencephalography. *Scientific Data*, 9(1), 2022. doi: 10.1038/s41597-022-01542-9. URL https://doi.org/10.1038/s41597-022-01542-9.

Maitreyee Wairagkar, Nicholas S. Card, Tyler Singer-Clark, Xianda Hou, Carrina Iacobacci, Lee M. Miller, Leigh R. Hochberg, David M. Brandman, and Sergey D. Stavisky. An instantaneous voice-synthesis neuroprosthesis. *Nature*, 644(8075):145–152, 2025. doi: 10.1038/s41586-025-09127-3. URL https://doi.org/10.1038/s41586-025-09127-3.

Francis R. Willett, Erin M. Kunz, Chaofei Fan, Donald T. Avansino, Guy H. Wilson, Eun Young Choi, Foram Kamdar, Matthew F. Glasser, Leigh R. Hochberg, Shaul Druckmann, Krishna V. Shenoy, and Jaimie M. Henderson. A high-performance speech neuroprosthesis. *Nature*, 620 (7976):1031–1036, 2023. doi: 10.1038/s41586-023-06377-x. URL https://doi.org/10.1038/s41586-023-06377-x.

Xueyi Zhang, Ruicong Wang, Jialu Sun, Siqi Cai, and Haizhou Li. Listening to the Brain: Multi-Band sEEG Auditory Reconstruction via Dynamic Spatio-Temporal Hypergraphs. In *Advances in Neural Information Processing Systems*. Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2025. doi: 10.52202/085713-4302. URL https://doi.org/10.52202/085713-4302.

A Appendix

A.1 One-Step Flow Matching as a Discriminative Model

Consider a network $\boldsymbol{v}_{\boldsymbol{\theta}}(\boldsymbol{y}_t, t, \boldsymbol{x})$ that takes as input an interpolated sample $\boldsymbol{y}_t \coloneqq (1-t)\boldsymbol{y}_0 + t\boldsymbol{y}_1$; the "time" of the interpolation, t; and a conditioning vector $\boldsymbol{x}$ (as well as parameters $\boldsymbol{\theta}$). In conditional flow matching, we train this network to predict (in the MMSE sense) the vector pointing toward the endpoint of the interpolation, $\boldsymbol{y}_1$:

$$\mathcal{L}_{\text{FM}}(\boldsymbol{\theta}) \coloneqq \left\langle \left\| \boldsymbol{v}_{\boldsymbol{\theta}}((1-T)\boldsymbol{Y}_0 + T\boldsymbol{Y}_1, T, \boldsymbol{X}) - (\boldsymbol{Y}_1 - \boldsymbol{Y}_0) \right\|^2 \right\rangle_{\boldsymbol{X}, \boldsymbol{Y}_0, \boldsymbol{Y}_1, T},$$

with interpolation time itself a random variable, $T \sim \mathcal{U}(0,1)$. Now consider a small variation on this loss that downweights the penalty for mispredicted velocities as they approach the target:

$$\mathcal{L}_{\text{alt}}(\boldsymbol{\theta}) \coloneqq \left\langle \left\| \boldsymbol{v}_{\boldsymbol{\theta}}((1-T)\boldsymbol{Y}_0 + T\boldsymbol{Y}_1, T, \boldsymbol{X}) - (1-T)(\boldsymbol{Y}_1 - \boldsymbol{Y}_0) \right\|^2 \right\rangle_{\boldsymbol{X}, \boldsymbol{Y}_0, \boldsymbol{Y}_1, T}.$$

Suppose we reparameterize this objective, without loss of generality, in terms of a different neural network, $\boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{y}_t, t, \boldsymbol{x})$, defined as

$$\begin{aligned} \boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{y}_t, t, \boldsymbol{x}) &\coloneqq \boldsymbol{v}_{\boldsymbol{\theta}}(\boldsymbol{y}_t, t, \boldsymbol{x}) + \boldsymbol{y}_t \\ &= \boldsymbol{v}_{\boldsymbol{\theta}}(\boldsymbol{y}_t, t, \boldsymbol{x}) + (1-t)\boldsymbol{y}_0 + t\boldsymbol{y}_1. \end{aligned}$$

The loss then becomes

$$\begin{aligned} \mathcal{L}_{\text{alt}}(\boldsymbol{\theta}) &= \left\langle \left\| \boldsymbol{\mu}_{\boldsymbol{\theta}}((1-T)\boldsymbol{Y}_0 + T\boldsymbol{Y}_1, T, \boldsymbol{X}) - (1-T)\boldsymbol{Y}_0 - T\boldsymbol{Y}_1 - (1-T)(\boldsymbol{Y}_1 - \boldsymbol{Y}_0) \right\|^2 \right\rangle_{\boldsymbol{X}, \boldsymbol{Y}_0, \boldsymbol{Y}_1, T} \\ &= \left\langle \left\| \boldsymbol{\mu}_{\boldsymbol{\theta}}((1-T)\boldsymbol{Y}_0 + T\boldsymbol{Y}_1, T, \boldsymbol{X}) - \boldsymbol{Y}_1 \right\|^2 \right\rangle_{\boldsymbol{X}, \boldsymbol{Y}_0, \boldsymbol{Y}_1, T}. \end{aligned}$$

That is, the modified flow-matching loss $\mathcal{L}_{\text{alt}}$ is equivalent to predicting the target, rather than the vector pointing to the target, at every point along the interpolation.

Now notice that if the losses, $\mathcal{L}_{\text{FM}}$ and $\mathcal{L}_{\text{alt}}$, were evaluated only at the initial time $t = 0$, they would coincide:

$$\mathcal{L}_0(\boldsymbol{\theta}) \coloneqq \left\langle \left\| \boldsymbol{v}_{\boldsymbol{\theta}}(\boldsymbol{Y}_0, 0, \boldsymbol{X}) - (\boldsymbol{Y}_1 - \boldsymbol{Y}_0) \right\|^2 \right\rangle_{\boldsymbol{X}, \boldsymbol{Y}_0, \boldsymbol{Y}_1} = \left\langle \left\| \boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{Y}_0, 0, \boldsymbol{X}) - \boldsymbol{Y}_1 \right\|^2 \right\rangle_{\boldsymbol{X}, \boldsymbol{Y}_0, \boldsymbol{Y}_1}.$$

Thus, imposing the flow-matching loss only at the initial time point is equivalent to requiring an arbitrary neural network $\boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{y}, t, \boldsymbol{x})$ to predict (in the MMSE sense) the output $\boldsymbol{y}_1$ from the inputs $\boldsymbol{x}$. Since a random noise vector $\boldsymbol{y}_0$ cannot facilitate this prediction, and since the time is fixed, $\boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{y}, t, \boldsymbol{x})$ will learn to ignore its first two argument, which we therefore omit (by abuse of notation) so that we can write the loss as

$$\mathcal{L}_0(\boldsymbol{\theta}) = \left\langle \left\| \boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{X}) - \boldsymbol{Y}_1 \right\|^2 \right\rangle_{\boldsymbol{X}, \boldsymbol{Y}_1},$$

which is simply the standard *discriminative* loss.

A.2 Additional Methodology and Implementation Details

A.2.1 Model Architecture

The neural encoder processes neural data at its original sampling rate, then aligns the encoded features with the mel-spectrogram frames (Table 4). Its convolutions are temporal 1-D convolutions, dilated by 2^i at layer i. T_{neural} and T_{mel} denote the number of neural samples and mel frames in a segment, respectively; on T15 the LPCNet feature frames share a 100 Hz grid with the neural samples, so the alignment is $1:1$.

The receptive field is therefore $1 + (k-1)\sum_{i<L} 2^i$ samples for kernel size k and L layers, and we match it in time rather than in depth: NZ0/NZ1 are sampled at 200 Hz with $L = 5$ (125 samples, 625 ms), T15 at 100 Hz with $L = 4$ (61 bins, 610 ms). A fifth layer on T15 would span 1.25 s, twice the 600 ms window the baseline of Wairagkar et al. (2025) is restricted to. The spans are also differently shaped: the NZ encoder is symmetrically padded and sees ± 312 ms, while the T15

Table 4: **Neural context encoder architecture.** The first convolutional block expands 64 input channels to 256. The next four blocks retain that width, with dilations of 2, 4, 8, and 16. A final projection produces 192 features per neural sample before alignment to mel frames.

Stage	Input shape	Output shape
Neural data input	$(B, 64, T_{\text{neural}})$	$(B, 64, T_{\text{neural}})$
Convolutional block 1	$(B, 64, T_{\text{neural}})$	$(B, 256, T_{\text{neural}})$
Conv1d ($64 \to 256$, $k = 5$, dilation 1)		
GroupNorm (8 groups)		
GELU		
Dropout ($p = 0.1$)		
Convolutional block ($\times 4$)	$(B, 256, T_{\text{neural}})$	$(B, 256, T_{\text{neural}})$
Conv1d ($256 \to 256$, $k = 5$, dilations $2, 4, 8, 16$ in order)		
GroupNorm (8 groups)		
GELU		
Dropout ($p = 0.1$)		
Feature projection: Conv1d ($k = 1$)	$(B, 256, T_{\text{neural}})$	$(B, 192, T_{\text{neural}})$
Linear interpolation	$(B, 192, T_{\text{neural}})$	$(B, 192, T_{\text{mel}})$
Transpose channel and time axes	$(B, 192, T_{\text{mel}})$	$(B, T_{\text{mel}}, 192)$
LayerNorm (192)	$(B, T_{\text{mel}}, 192)$	$(B, T_{\text{mel}}, 192)$

Table 5: **Diffusion Transformer (DiT) architecture.** The table shows the time-embedding branch, one of eight identical DiT blocks, and the output head. The time embedding supplies the AdaLN modulation in each block.

Stage	Input shape	Output shape
Frame-wise concatenation, $\mathbf{x}_t$ and $\mathbf{c}$	$(B, T_{\text{mel}}, 80 + 192)$	$(B, T_{\text{mel}}, 272)$
Input projection (linear)	$(B, T_{\text{mel}}, 272)$	$(B, T_{\text{mel}}, 384)$
Time-embedding branch	$(B,)$	$(B, 384)$
Sinusoidal embedding	$(B,)$	$(B, 384)$
Linear + SiLU	$(B, 384)$	$(B, 1536)$
Linear	$(B, 1536)$	$(B, 384)$
DiT block ($\times 8$)	$(B, T_{\text{mel}}, 384)$	$(B, T_{\text{mel}}, 384)$
AdaLN-Zero modulation (SiLU + linear)	$(B, 384)$	$(B, 2304)$
AdaLN (attention path)	$(B, T_{\text{mel}}, 384)$	$(B, T_{\text{mel}}, 384)$
Multi-head self-attention (6 heads, RoPE)	$(B, T_{\text{mel}}, 384)$	$(B, T_{\text{mel}}, 384)$
Gated attention residual addition		
AdaLN (feed-forward path)	$(B, T_{\text{mel}}, 384)$	$(B, T_{\text{mel}}, 384)$
Feed-forward network (GEGLU, hidden width 1536)	$(B, T_{\text{mel}}, 384)$	$(B, T_{\text{mel}}, 384)$
Gated feed-forward residual addition		
Output head	$(B, T_{\text{mel}}, 384)$	$(B, T_{\text{mel}}, 80)$
Output modulation (SiLU + linear)	$(B, 384)$	$(B, 768)$
AdaLN	$(B, T_{\text{mel}}, 384)$	$(B, T_{\text{mel}}, 384)$
Linear: velocity projection (zero-initialized)	$(B, T_{\text{mel}}, 384)$	$(B, T_{\text{mel}}, 80)$

encoder is causal—left-padded, and normalised over channels at each time step rather than over the segment—so frame k sees $[k - 60, k]$, 610 ms strictly in the past. Table 4 features the NZ-variant of the neural encoder architecture.

The DiT concatenates the current 80-band mel state with the 192-dimensional neural context at each frame (Table 5). A sinusoidal embedding of flow time modulates all eight transformer blocks through adaLN-Zero. Each block contains six-head self-attention with rotary positional embeddings (RoPE), followed by a GEGLU feed-forward network with a fourfold hidden expansion. The DiT predicts an 80-dimensional velocity at each mel frame.

A.2.2 Model Training

On NZ0/NZ1, we train our models on randomly sampled 4-second segments ($L_{\text{seg}} = 4$ s) for 40,000 steps with a batch size of 16. We use AdamW with a peak learning rate of 2×10^{-4} and $\beta_1 = 0.9$, $\beta_2 = 0.95$. We use cosine schedule for the learning rate with warmup period of 1000 steps. Model weights are averaged throughout training using an exponential moving average (EMA) with decay 0.999. We use these EMA weights for validation and test evaluation.

We measure validation MCD every 2,000 steps. Because MCD can change substantially between successive validations, we smooth it with a separate EMA with decay 0.8. Training always continues to 40,000 steps; among checkpoints evaluated after step 5,000, we select the one with the lowest smoothed validation MCD.

On T15, we keep the segment length ($L_{\text{seg}} = 4$ s, i.e. 400 frames at the dataset's 100 Hz bin rate), the optimizer (AdamW, peak learning rate 2×10^{-4}, $\beta_1 = 0.9$, $\beta_2 = 0.95$), the cosine schedule with a 1000-step warmup, and the EMA of the weights (decay 0.999, used for all validation and test evaluation). Weight decay is 0.01 and gradients are clipped to unit norm. The batch size is raised to 128, and the decoder is made causal: the neural encoder is left-padded so that frame k reads only $[k - 60, k]$ (610 ms of past context, matching the baseline's information set), its normalization is applied per frame rather than pooled over the sequence, and the velocity network's self-attention is restricted by a causal mask. No session-conditioning parameters are used. Targets are smoothed within each trial by a Gaussian of $\sigma = 2$ frames during training only (same targets used by Wairagkar et al.); evaluation always scores against the unsmoothed targets.

T15 is evaluated under the protocol of Wairagkar et al. (2025), in which each of the eight evaluation days is decoded by a model trained on every session recorded before it. We therefore train a chain of eight stages ratheirst stage is trained from scratch on the 29 sessions preceding day 172 for 36,000 steps; each subsequent stage warm-starts from the previous s, and trains for a further 10,000 steps with the warmup shortened to 200 steps, since it begins from converged weights.

We measure validation feature PCC every 1,000 steps on 50 trials held out from the most recent training session, computed frame-aligned over speech frames only using the released word boundaries. Unlike MCD on NZ, this quantity is stable between successive validations, so it is used unsmoothed and without a minimum-step floor; we select the checkpoint with the highest validation feature PCC.

A.2.3 Sample generation

The velocity field is a transformer over the time axis and is trained on 4 s crops, whereas held-out trials are typically much longer. We therefore integrate the ODE in overlapping windows of exactly the training length. With a hop of $H = W/2$, the window onsets are

$$\mathcal{S} \;=\; \{\, iH \;:\; i = 0, 1, \ldots,\; iH + W \le T \,\} \;\cup\; \{\, T - W \,\}, \tag{5}$$

the final term anchoring the last window to the end of the trial so that the tail is covered without padding. Each window is integrated independently from its own slice of the source (see Section 3.2.3), and the results are combined by overlap-add under a Hann taper $w(n) = \frac{1}{2}\big(1 - \cos \frac{2\pi n}{W-1}\big)$,

$$\hat{x}_1(t) \;=\; \frac{\sum_{s \in \mathcal{S}} w(t - s)\, x_1^{(s)}(t)}{\sum_{s \in \mathcal{S}} w(t - s)}, \qquad t \in [s,\, s + W), \tag{6}$$

where dividing by the accumulated weight makes the reconstruction exact wherever the windows overlap. Only the ODE solve is windowed: the source x_0 and the conditioning sequence $c = E(s)$ are computed once over the whole trial, so the convolutional context encoder retains its full temporal context and no information is lost at a window boundary. Trials shorter than W are integrated in a single pass. All reported metrics are computed on the reconstructed full-length trial.

A.3 Full Data-Processing Chain for the NZ datasets

A.3.1 Experimental protocol

Sentences presented to the participants were harvested from the `text_spok` sub-corpus of the Corpus of Contemporary American English, a collection of conversational English derived primarily

from television and radio public-affairs programs. From this sub-corpus, all sentences consisting exclusively of the 1,000 most frequent words were extracted, yielding approximately 3,000 candidate sentences. In each trial, a sentence was randomly selected from this corpus and visually presented to the participant, who rendered it aloud twice: first by reading it directly, and subsequently from memory after the text disappeared. Trials were organized into blocks of 40. The two participants completed 8 and 9 blocks, yielding 320 and 360 sentences (comprising 537 and 557 unique words, respectively). In total, we collected 33 and 28 minutes of audio data along with corresponding sEEG recordings from both participants.

A.3.2 LINE-NOISE REMOVAL

All channels were labeled according to their putative physiological locations, and channels containing non-sEEG data were excluded prior to preprocessing. We first removed temporal outliers from the recordings, followed by notch filtering at 60 Hz and its harmonics up to the Nyquist frequency to attenuate line noise. Finally, channels exhibiting large artifacts or excessive noise were identified through visual inspection and removed from further analysis.

A.3.3 REFERENCING

We applied common average referencing (CAR) within each depth electrode. CAR assumes that signals shared across multiple contacts primarily reflect global artifacts or non-local activity. For each depth electrode, we therefore computed the mean signal across all available contacts and subtracted this mean from each individual contact. Because sEEG recordings capture highly localized neural activity, applying CAR globally across electrodes may attenuate or redistribute focal physiological signals, as well as propagate localized mechanical artifacts to otherwise unrelated channels. Restricting the reference to contacts within the same depth electrode helps preserve the spatial specificity of the recorded neural activity.

A.3.4 HIGH-γ COMPUTATION

We then extracted high-γ activity following the preprocessing procedure described by Makin et al. (Makin et al., 2020). Each signal was passed through log-spaced, Gaussian-shaped bandpass filters spanning the high-γ range of 70–200 Hz. The analytic amplitude (envelope) was subsequently computed for each filtered signal, and the resulting envelopes were averaged to obtain a single time-varying signal, which we refer to as the high-γ signal.

A.3.5 LOCAL FIELD POTENTIALS (LFPS)

Previous work has demonstrated that incorporating local field potentials (LFPs) can improve speech synthesis performance Anumanchipalli et al. (2019). To extract these LFP features, we applied a fourth-order Butterworth low-pass filter (cutoff at 50 Hz) to the CAR-processed signals. Both the resulting LFP signals and the high-γ signals were then downsampled to 200 Hz and concatenated channel-wise, yielding an input representation with twice the number of channels.

A.3.6 ELECTRODE SELECTION

Electrode placement in clinical sEEG recordings is primarily determined by clinical considerations, such as the localization of epileptogenic regions, and consequently, many implanted electrodes may not capture neural activity directly relevant to speech. Although a sufficiently large training set may allow a neural network to learn to disregard such channels, this is particularly challenging in the small-data regime ($\sim$30 minutes) considered in this work. We therefore perform an initial, data-driven electrode-selection procedure independently for each participant in the NZ datasets.

Specifically, we train a three-layer temporal convolutional neural network (CNN) on the same speech synthesis task and use input gradients to estimate the saliency, or importance, of each channel Simonyan et al. (2014); Makin et al. (2020). For an input sEEG–mel-spectrogram pair $(\mathbf{X}^{(n)}, y^{(n)})$, we define the importance of channel c as

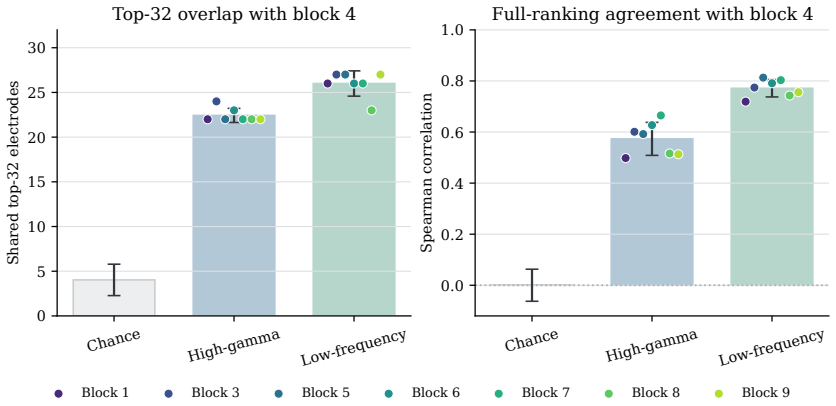


Figure 4: **Salient-electrode agreement with NZ0 block 4.** Each colored point is one of the other seven held-out blocks; its color identifies the same block in both panels. Bars show means and error bars show one standard deviation across those seven comparisons. The gray chance bars show the mean and standard deviation of 100,000 independent random rankings of the same 254 retained electrodes (seed 42). The top-32 sets are taken from these rankings. The exact central 95% chance range for top-32 overlap is 1–8 electrodes.

$$I_c := \frac{1}{N}\sum_{n=1}^{N}\frac{1}{T}\sum_{t=1}^{T}\left|\frac{\partial \mathcal{L}(f_\theta(\mathbf{X}^{(n)}), y^{(n)})}{\partial x_{t,c}^{(n)}}\right|, \tag{7}$$

where $\mathcal{L}$ denotes the mean squared error loss, $f_\theta(\cdot)$ is the aforementioned trained CNN, and t, c, and n denote the time step, channel index, and trial index, respectively. The gradients are obtained by backpropagating the loss through the entire network to the input space, thereby quantifying the sensitivity of the model's output error to perturbations in each input channel Simonyan et al. (2014). We take the absolute value of the gradient so that both positive and negative sensitivities contribute to the estimated channel importance. The resulting values are averaged across time and trials to identify channels exhibiting consistently strong influence on the model's predictions.

Because our input representation comprises both high-γ and low-frequency (LFC) features, each physiological contact contributes two input channels. We independently rank these channels according to their importance scores and retain the 32 highest-scoring channels from each feature type. This procedure yields a total of 64 selected input channels for subsequent speech-synthesis experiments for both our proposed flow-based method and also each of the benchmark models.

Since the data used in this electrode-selection procedure were subsequently used to train and test models, there is theoretically some information leakage from the training into the test set. However, the effects will be very small in practice, since electrode selection was made on one fold, whereas results are cross-validated. Furthermore, the same set of 64 channels was used for all algorithms, so any upward bias would affect all algorithms equally—the comparisons would remain fair.

To check whether the selection depends strongly on block 4, we repeated the analysis with each of the other held-out blocks of NZ0. The left panel of Fig. 4 compares each block's top-32 electrodes with those selected using block 4. On average, 22 high-γ electrodes and 26 low-frequency (LFC) electrodes overlap, compared with about 4 expected by chance. The right panel compares the rankings of all 254 electrodes using Spearman correlation. Both feature types show correlations well above the chance level of approximately zero. These results suggest that the selected electrodes are fairly stable across held-out blocks.

A.4 Additional Results

A.4.1 Probability Paths for Flow Matching

We compare the linear (OT) interpolant used in our finalized flow model against the variance-preserving (VP) and variance-exploding (VE) paths of Lipman et al. (2023) (VE on the usual ge-

Table 6: **Probability path and coupling.** NZ0, held-out block 4, mean ± s.d. over 3 training seeds (42, 123, 7); everything else, including the training budget, is held fixed. Rows 1, 3 and 4 share the minibatch-OT coupling and change only the interpolant; row 2 keeps the linear path and pairs x_0 with x_1 by index instead of by the minibatch assignment. Each path is evaluated with the shipped 8-step Euler solve and with a single step, on the same checkpoint and the same x_0. S_{rf} is the rectified-flow straightness functional (Liu et al., 2023) and dev. the trajectory deviation of Chen et al. (2024a); both are 0 for a perfectly straight flow, are properties of the arm rather than of N, and are measured on an RK4 reference trajectory (see text). Best in **bold**, second best underlined.

Variant	N	ESTOI ↑	UTMOS ↑	SpkSim ↑	MCD ↓	S_{rf} ↓	dev. % ↓
Ours: linear (OT coupling)	8	0.403 ±0.000	1.903 ±0.005	**0.717** ±0.008	6.892 ±0.097	**0.0046**	**1.67**
	1	0.395 ±0.004	1.743 ±0.054	0.691 ±0.003	6.825 ±0.085	–	–
linear (independent coupling)	8	**0.408** ±0.004	1.882 ±0.065	0.715 ±0.002	6.949 ±0.085	0.0048	1.70
	1	0.399 ±0.005	1.743 ±0.026	0.684 ±0.003	6.832 ±0.053	–	–
VP diffusion (Lipman et al., 2023)	8	0.392 ±0.013	**1.980** ±0.023	0.697 ±0.004	**6.803** ±0.100	0.4684	14.67
	1	0.048 ±0.002	1.842 ±0.004	-0.047 ±0.000	10.936 ±0.002	–	–
VE diffusion (Lipman et al., 2023)	8	0.357 ±0.004	1.589 ±0.157	0.527 ±0.012	7.345 ±0.257	2.9879	36.29
	1	0.076 ±0.002	1.092 ±0.035	-0.029 ±0.008	11.946 ±0.162	–	–

ometric schedule, $\sigma \in [10^{-2}, 10]$), holding the field, the conditioning, the coupling and the solver fixed. We employ two separate measures to assess how straight the resulting flows are for each interpolant. First, we use the rectified-flow straightness functional (Liu et al., 2023), which measures the departure of the velocity along the solved path from the constant straight-line velocity according to

$$S_{\mathrm{rf}} = \frac{\int_0^1 \mathbb{E}\big\|(x_1 - x_0) - v_\theta(x_t, t, c)\big\|^2 \, dt}{\mathbb{E}\|x_1 - x_0\|^2}, \tag{8}$$

Second, we also use the trajectory deviation of Chen et al. (2024a), which measures the same departure in state space, as the excursion of the path from the line through its two endpoints, relative to the distance between them. It is defined as:

$$\Delta = \frac{\mathbb{E}_t \, \mathbb{E}\big[\, d\big(x_t, \, [x_0 \, x_1]\big)\big]}{\mathbb{E}\|x_1 - x_0\|}, \qquad d\big(x_t, \, [x_0 \, x_1]\big) = \big\|x_t - \big((1-t)x_0 + t\, x_1\big)\big\|, \tag{9}$$

where both expectations are taken over trials before the ratio, as in Chen et al. (2024a). Both measures vanish exactly for a straight flow; we evaluate them on an RK4 reference trajectory integrated window-by-window as at inference. They order the paths exactly as quality does, and they predict what happens when the solver budget is cut. A single Euler step costs the near-straight linear flow little (ESTOI .403 → .395), because for a straight path one step *is* the exact solve; on the curved paths the same cut is catastrophic (VP .392 → .048, VE .357 → .076, both with negative speaker similarity). The minibatch-OT coupling (Tong et al., 2024), by contrast, does not affect the speech quality: an independent pairing matches it on every metric (ESTOI .408 vs. .403) and on both straightness measures of Eqs. (8) and (9).

A.4.2 Sampling Variance

The flow model can generate different spectrograms for the same neural recording. To measure this variation, we generated $K = 128$ samples per test trial on all eight NZ0 folds. This gives 160 test trials and 20,480 scored samples in total. We scored each sample separately, then averaged the scores over samples, trials, and folds. It is important to note that we did not average the spectrograms before scoring. The *single draw* row in Table 8 shows the mean score across the draws. The *within-trial s.d.* shows how much a trial's score changes when we draw a different sample. This measures sampling variation, not the variation across folds. The *95% of draws* row gives the central interval of a trial's own draws, which is easier to read than a standard deviation: a single trial's ESTOI covers $[0.339, 0.404]$ and its UTMOS covers $[1.588, 2.179]$. We report both rows at $K = 8$ and at $K = 128$. The standard deviation is the same at either setting (0.017 ESTOI), so drawing more samples does not make the conditional distribution wider.

Selecting the sample with the highest reference ESTOI gives an improvement of $+0.043$ ESTOI compared to the Monte Carlo average score. However, it is not available at inference because it requires the reference speech. Selecting by UTMOS is possible without a reference, but it does

Table 7: **Effect of ODE evaluations and sample averaging on NZ0 block 4**. Results use one training seed (42); values are mean ± s.d. across trials. All rows use the same neural-conditioned linear-OT flow model and Euler solver. $K = 1$ evaluates one sample per trial; $K = 8$ averages eight generated mel spectrograms before vocoding and scoring. NFE is the number of velocity-field evaluations per draw.

Estimator	r_{sp} ↑	SpkSim ↑	ESTOI ↑	UTMOS ↑	MCD ↓
NFE=8, $K = 1$	0.806 ±0.047	**0.711** ±0.058	0.403 ±0.090	**1.909** ±0.206	7.167 ±1.097
NFE=1, $K = 1$	0.811 ±0.045	0.687 ±0.058	0.395 ±0.090	1.750 ±0.194	**7.072** ±1.027
NFE=8, $K = 8$	0.812 ±0.047	0.699 ±0.058	**0.415** ±0.090	1.780 ±0.273	7.099 ±1.071
NFE=1, $K = 8$	**0.814** ±0.045	0.686 ±0.062	0.403 ±0.086	1.685 ±0.160	7.074 ±1.032

Table 8: **Sampling variance on NZ0.** Our decoder returns a sample, so we draw K independent x_0 per trial on all 8 folds (160 trials). *Single draw* is the Monte-Carlo average over the K draws — each scored separately, the scores averaged — which is the estimator reported in Table 1 and equals the expected score of one random draw. *Within-trial s.d.* is how far a single trial's score moves with the draw; *95% of draws* is the central interval of that trial's own draws ($p_{2.5}$ to $p_{97.5}$). Both are averaged over trials and given at $K = 8$ and $K = 128$. The *best-of-K* rows pick one draw per trial by the named score at $K = 128$, with the change from the single draw in parentheses. Selecting on ESTOI reads the ground truth, so its $+0.043$ is headroom rather than a result; selecting on reference-free UTMOS is realisable and captures none of it (-0.001 ESTOI).

Estimator	r_{sp} ↑	SpkSim ↑	ESTOI ↑	UTMOS ↑	MCD ↓
single draw	0.803	0.648	0.372	1.882	6.528
within-trial s.d. (K=8)	0.006	0.022	0.017	0.141	0.180
within-trial s.d. (K=128)	0.006	0.023	0.017	0.156	0.184
95% of draws (K=8)	[0.794, 0.810]	[0.617, 0.677]	[0.349, 0.396]	[1.676, 2.062]	[6.283, 6.774]
95% of draws (K=128)	[0.792, 0.813]	[0.602, 0.689]	[0.339, 0.404]	[1.588, 2.179]	[6.185, 6.880]
best-of-K (ESTOI), K=128	0.806 (+0.003)	0.647 (0.000)	0.415 (+0.043)	1.875 (−0.007)	6.435 (−0.093)
best-of-K (UTMOS), K=128	0.802 (−0.001)	0.648 (+0.001)	0.371 (−0.001)	2.292 (+0.410)	6.501 (−0.026)

not improve ESTOI here: it raises UTMOS by $+0.410$ while leaving ESTOI at -0.001. Within a trial, the correlation between changes in ESTOI and UTMOS is $+0.025$, which is detectable at this sample size but far too small to be useful. The corresponding correlations are -0.221 for MCD and $+0.231$ for r_{sp}, but both of these metrics require the reference.

A.4.3 Effect of Averaging Generated Spectrograms

We examine how the number of Euler steps and averaging generated spectrograms affect fidelity and perceptual quality. With one step, the model makes a single update from Gaussian noise, which is very close to a discriminative modeling case; with more steps, it can follow the learned velocity field and refine the sample. For each setting, we either evaluate one sample or generate eight samples and average their mel spectrograms before vocoding. This average estimates the mean of the model's generated spectrograms for a given neural input. Note that here we are averaging the generated spectrograms, which gives us an estimate of the mean of the target distribution. This is different from the main results where we evaluate multiple samples separately and then average the evaluation scores.

In short, we have two ways to move towards the conditional mean: i) use fewer NFEs, and ii) average predicted spectrograms (by tuning K). Table 7 depicts results from the full matrix. We observe that if we keep K fixed at 1 and reduce NFEs, then r_{sp} and MCD both marginally improve, however both perceptual metrics degrade (SpkSim and UTMOS). We achieve a similar outcome if we keep NFEs fixed at 8 and increase K from 1 to 8 (first and third rows of Table 7). If we decide to vary both NFEs and K simultaneously, we observe the largest effect, the final row depicts this. This setting achieves the highest r_{sp} and lowest set of UTMOS and SpkSim scores, suggesting a closer proximity to the conditional mean. These observations agree closely with the distortion–perception trade-off depicted in Fig. 3.

Table 9: **Effect of the neural encoder's receptive field.** NZ0, held-out block 4; mean ± s.d. over three training seeds (42, 123, 7). The first column lists the five layer dilations and the resulting receptive field; a^i denotes dilations $a^0, \ldots, a^4$. All rows except the last use the same 1.45 M-parameter encoder and training budget. The last row keeps the 625 ms receptive field but reduces the encoder to 0.22 M parameters. Best values are **bold**; second best are underlined.

Variant	ESTOI ↑	UTMOS ↑	SpkSim ↑	MCD ↓
1^i (105 ms)	0.382 ±0.011	1.898 ±0.040	0.700 ±0.006	6.959 ±0.139
$1, 1, 2, 4, 4$ (245 ms)	<u>0.405</u> ±0.005	1.857 ±0.018	0.706 ±0.002	6.939 ±0.125
$1, 1, 4, 4, 8$ (365 ms)	0.403 ±0.003	1.868 ±0.046	0.714 ±0.011	7.008 ±0.079
$1, 4, 4, 8, 8$ (505 ms)	0.404 ±0.005	<u>1.902</u> ±0.028	<u>0.716</u> ±0.007	7.024 ±0.077
Ours: 2^i (625 ms)	0.403 ±0.000	**1.903** ±0.005	**0.717** ±0.008	**6.892** ±0.097
$1, 2, 4, 16, 16$ (785 ms)	0.405 ±0.005	1.896 ±0.013	0.706 ±0.009	6.978 ±0.071
$2, 4, 8, 16, 20$ (1.0 s)	**0.407** ±0.009	1.857 ±0.024	0.714 ±0.002	<u>6.929</u> ±0.096
4^i (6.8 s)	0.363 ±0.007	1.825 ±0.030	0.680 ±0.003	7.454 ±0.058
2^i (625 ms), 0.22 M params	0.383 ±0.012	1.862 ±0.047	0.697 ±0.012	7.181 ±0.182

Table 10: **Transcription error rates on the NZ datasets.** Word and character error rate under leave-one-block-out cross-validation (**NZ0**: 8 folds; **NZ1**: 9 folds). Entries are the median [interquartile range] of the per-trial rate over all held-out trials ($n = 160$ and $n = 180$). Markers on our row are a one-sided Wilcoxon signed-rank test on the per-fold corpus rates, paired over folds against VocalMind (He et al., 2025) ($^{***}\, p < 0.005$, $^{**}\, p < 0.01$, $^{*}\, p < 0.05$, † tested, not significant).

	NZ0		NZ1	
Model	WER ↓	CER ↓	WER ↓	CER ↓
VocalMind (He et al., 2025)	<u>1.00 [0.83, 1.11]</u>	<u>0.72 [0.63, 0.85]</u>	<u>1.00 [0.81, 1.12]</u>	<u>0.72 [0.62, 0.84]</u>
Wairagkar et al. (Wairagkar et al., 2025)	1.00 [0.94, 1.19]	0.79 [0.70, 0.95]	1.00 [0.93, 1.12]	0.77 [0.70, 0.91]
Ours	**0.88 [0.66, 1.09]***	**0.62 [0.48, 0.77]****	**0.97 [0.70, 1.12]****	**0.67 [0.51, 0.83]***

A.4.4 Additional Ablation Studies on NZ

Effect of Receptive Field Table 9 compares different dilation patterns in the neural encoder. Our 625 ms setting gives the best UTMOS, speaker similarity, and MCD, while its ESTOI is close to the best value. A 1 s receptive field slightly improves ESTOI, but not the other metrics. Extending the receptive field to 6.8 s worsens all four metrics. Reducing the encoder size while keeping the 625 ms receptive field also hurts performance, so capacity matters as well.

Transcription Error Rates on NZ0 and NZ1 Table 10 shows word and character error rates for our proposed flow based method and two of the strongest benchmarks on the NZ datasets (Vocalmind and Wairagkar et al.) as per the speech metrics described in the main paper. Since the participants' data are protected under IRB protocol, it was not possible to employ human evaluation services, hence we used the Whisper ASR model for transcription purposes. Our model achieves the lowest median WER and CER on both participants, and compared against the stronger baseline, we achieve a significantly lower set of rates under a one-sided signed rank test. These error rates remain high, so we do not use them as evidence that the model recovers the exact spoken words.

Tuning Wairagkar et al. on NZ0 Table 11 tests several changes to the Wairagkar et al. baseline. Adding 200–400 ms of future context improves ESTOI and speaker similarity over the causal setting. Adding more past context alone does not give the same benefit. The original model uses a causal window for real-time use. In the NZ recordings, speech is audible, so some neural responses may follow the sound. Future context may help capture these responses. The optimization changes have mixed effects. For example, L1 loss gives the best speaker similarity and MCD in its group, but lower ESTOI than the MSE baseline. Larger models also do not improve every metric. The best setting depends on which metric we consider.

Table 11: **Tuning the Wairagkar et al. (2025) baseline on NZ0.** Results for held-out block 4 with training seed 42, evaluated with DTW alignment. The rows test context, optimization, and model capacity. Each group has its own baseline (*) and training settings, so compare rows only within a group. Best values within each group are **bold**.

Variant	Params	PCC_{sp} ↑	SpkSim ↑	ESTOI ↑	UTMOS ↑	MCD ↓
Context (8k iters, batch 16)						
600 ms causal *(paper)**	3.44M	0.710	0.281	0.202	1.139	6.491
+ 200 ms lookahead	3.44M	0.793	0.371	0.326	**1.521**	6.906
+ 400 ms lookahead	3.44M	**0.805**	**0.395**	**0.327**	1.462	7.028
+ 600 ms lookahead	3.44M	0.804	0.347	0.308	1.512	6.934
1200 ms causal *(token-matched control)*	3.44M	0.702	0.286	0.185	1.166	**6.330**
Optimization (8k iters, batch 16, 400 ms lookahead)						
dropout 0.5, lr 10^{-4}, MSE*	3.44M	0.807	0.375	0.302	1.503	6.777
dropout 0.35	3.44M	0.807	0.373	**0.317**	1.361	6.864
dropout 0.65	3.44M	0.795	0.313	0.273	1.487	7.063
weight decay 10^{-2}	3.44M	0.806	0.368	0.299	1.485	6.807
weight decay 10^{-1}	3.44M	**0.810**	0.365	0.299	1.501	6.789
lr 3×10^{-4}	3.44M	0.796	0.318	0.310	1.440	6.914
lr 3×10^{-5}	3.44M	0.789	0.264	0.252	1.425	6.780
L1 loss	3.44M	0.792	**0.404**	0.289	1.349	**6.647**
1200 ms past + 400 ms future	3.44M	0.791	0.295	0.281	1.342	7.190
4 blocks	2.12M	0.794	0.321	0.301	**1.508**	6.768
Capacity (8k iters, batch 8, 400 ms lookahead)						
h=128, 8 blocks *(settled config)**	3.44M	0.803	0.357	0.329	1.492	6.966
12 blocks	4.76M	0.811	0.368	0.329	1.406	6.776
h=192	6.83M	**0.813**	0.365	0.337	1.489	6.881
h=256	11.52M	0.802	**0.386**	0.322	1.507	6.885
h=256, MLP $2048/1024 \approx$ VocalMind	13.40M	0.804	0.337	**0.343**	**1.599**	6.655
h=256, 12 blocks	16.79M	0.811	0.367	0.340	1.504	6.708
h=384 $\approx$ ours	24.85M	0.804	0.321	0.316	1.449	**6.540**

A.4.5 Full Evaluation Suite for T15

Table 12 compares our model with the released online audio. Our model has a lower WER (0.22 versus 0.56). It also improves DTW-aligned Pearson correlation (speech frames only), MCD, ESTOI, UTMOS, and speaker similarity. The frame-aligned speech correlation is the exception: it is lower for our model (0.570 versus 0.658). This shows why both aligned and unaligned scores are useful. Fig. 5 shows the WER, DTW-aligned PCC, and MCD for each evaluation day, rather than only the pooled result. This also shows that our results are consistently better than the Wairagkar et al. (2025) across all the test days.

A.5 Qualitative Results

Figs. 6 and 7 are predicted spectrograms from our proposed model alongside the best benchmark and the ground truth for the NZ datasets. Prior to plotting the spectrograms, we applied a random pitch-shift as in (Verwoert et al., 2022) to anonymize each of the participants' voices. We also present the transcriptions generated by Whisper after vocoding the same spectrograms through the HiFi-GAN vocoder.

We also present similar results for the T15 dataset in Fig.8, however since the dataset has LPCNet features, we transform the predicted features to speech waveforms first and then compute the mel spectrograms. Although the dataset is publicly available, we still pitch-shift it to anonymize the participant's voice prior to presenting the spectrograms.

For the NZ datasets, we observe that the benchmark shows blurring behavior in the spectrograms, especially toward the high frequency bins. Overall our method achieves much sharper predictions and overall better speech transcriptions. When contrasting with qualitative and quantitative performance metrics, it can be stated that the synthesis quality in the T15 dataset is much superior, partly due to the higher data volume, but probably more so due to the recording quality.

Table 12: **Full evaluation on T15 (Wairagkar et al., 2025).** Results for 964 trials from eight evaluation days (days 172–195). The Wairagkar et al. column uses the released online audio. Both systems use the same LPCNet reference for reference-based metrics. Our model uses 8-step Euler sampling with $K = 8$ independent draws per trial. WER is measured by Whisper `medium.en` and reported as median [IQR] over pooled draws; other metrics are means ± s.d. over trials after averaging the draw scores. The last column is the median, across trials, of the within-trial s.d. over our model's $K = 8$ draws. Pearson correlation on speech frames (r_{sp}) and ESTOI are shown with (default for T15) and without DTW; MCD is shown with and without inter-word silences. UTMOSv2 is reference-free; SpkSim uses ECAPA-TDNN embeddings. Bold marks the best value. The CNN-DiT regressor column is our decoder with the flow objective replaced by an MSE regression onto the target features, at an identical architecture and parameter count. *** marks a paired, one-sided Wilcoxon test against the online released audio from Wairagkar et al. with $p < 0.005$.

Metric	Wairagkar et al.	CNN-DiT regressor	**Ours**	Draw s.d.
WER ↓	0.56 [0.29–0.83]	0.22 [0.00–0.44]	**0.22 [0.00–0.40]*****	0.10
$r_{sp,DTW}$ ↑	0.850 ± 0.039	0.890 ± 0.036	**0.898 ± 0.037*****	0.010
r_{sp} (frame-aligned) ↑	**0.658 ± 0.084**	0.604 ± 0.096	0.570 ± 0.095	0.023
MCD w/o sil. (dB) ↓	6.73 ± 0.66	5.57 ± 0.64	**5.43 ± 0.70*****	0.18
MCD w/ sil. (dB) ↓	2.96 ± 0.53	2.43 ± 0.45	**2.37 ± 0.45*****	0.08
ESTOI ↑	0.177 ± 0.057	**0.210 ± 0.066**	0.204 ± 0.066***	0.019
$ESTOI_{DTW}$ ↑	0.253 ± 0.055	0.355 ± 0.064	**0.404 ± 0.069*****	0.018
UTMOS ↑	1.73 ± 0.16	1.88 ± 0.16	**1.90 ± 0.10*****	0.13
SpkSim ↑	0.741 ± 0.074	0.801 ± 0.063	**0.835 ± 0.051*****	0.019

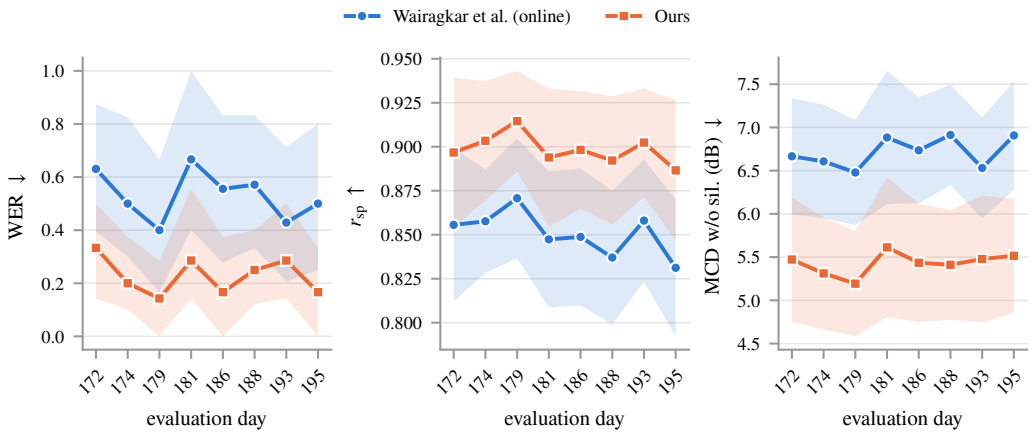


Figure 5: **Synthesis quality across T15 evaluation days (Wairagkar et al., 2025).** For each day, we use a model trained on all earlier sessions, as in the original protocol. The eight days contain 112, 110, 131, 131, 132, 132, 86, and 130 trials. Points show the daily median WER, mean Pearson correlation (r_{sp}, speech frames only), and MCD without inter-word silences. Shaded bands show the WER interquartile range or ± one s.d. for r_{sp} and MCD. We compare the released online audio with our causal flow model on the same trials and reference. Our model uses 8-step Euler and $K = 8$ draws per trial; scores are averaged over draws, while WER uses the median over pooled draws.

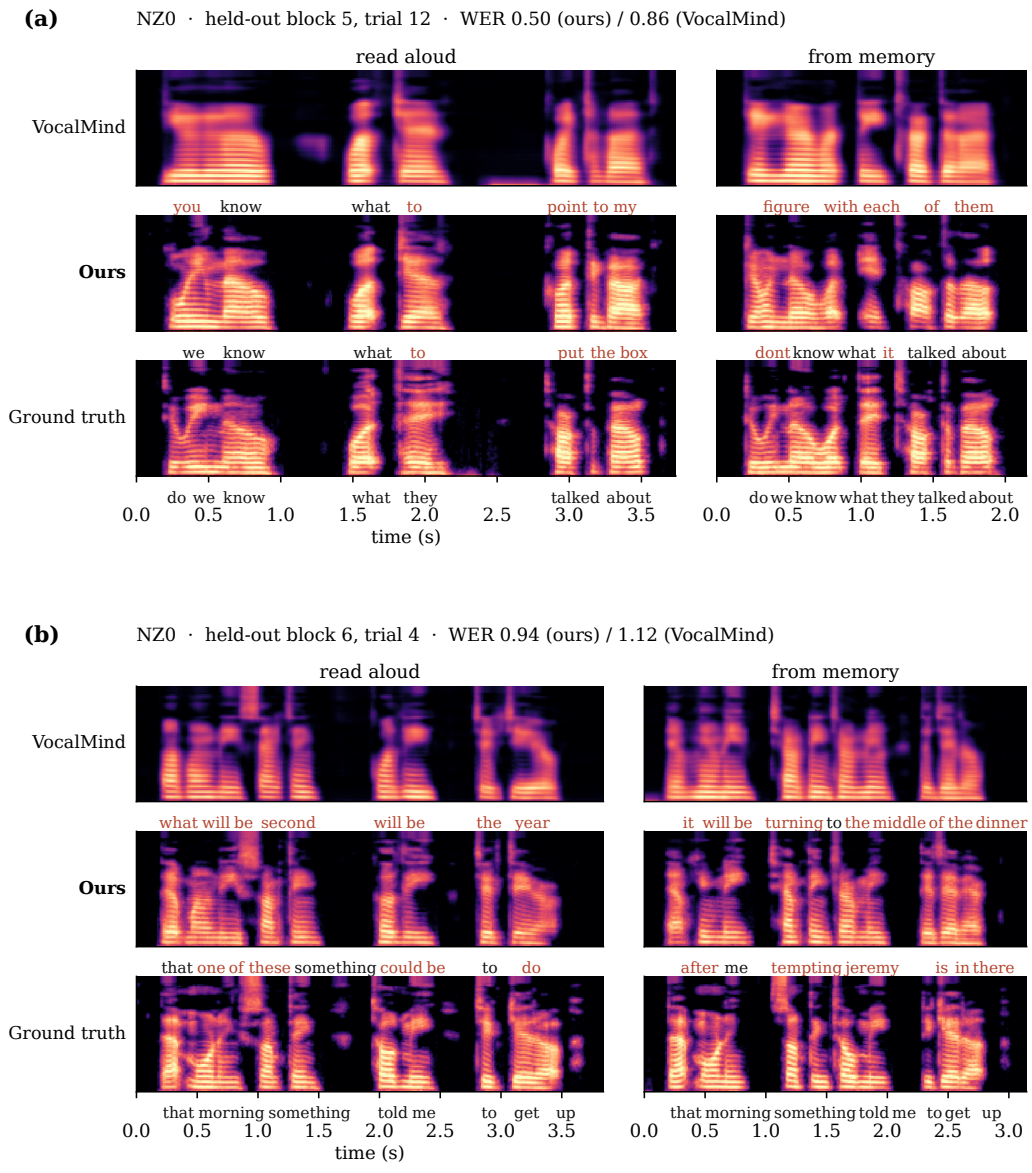


Figure 6: **Held-out trials decoded by the strongest published baseline and by our method on NZ0.** Each panel is one test trial of one leave-one-block-out fold; rows share a color scale set by the ground truth, and all spectrograms are pitch-shifted by a common offset for de-identification, which leaves the time and mel axes unchanged. Beneath each row is the Whisper transcription of that row's vocoded audio. We used the reference as Whisper's transcription of the vocoded ground truth, and words are printed in red where the predictions disagree with it.

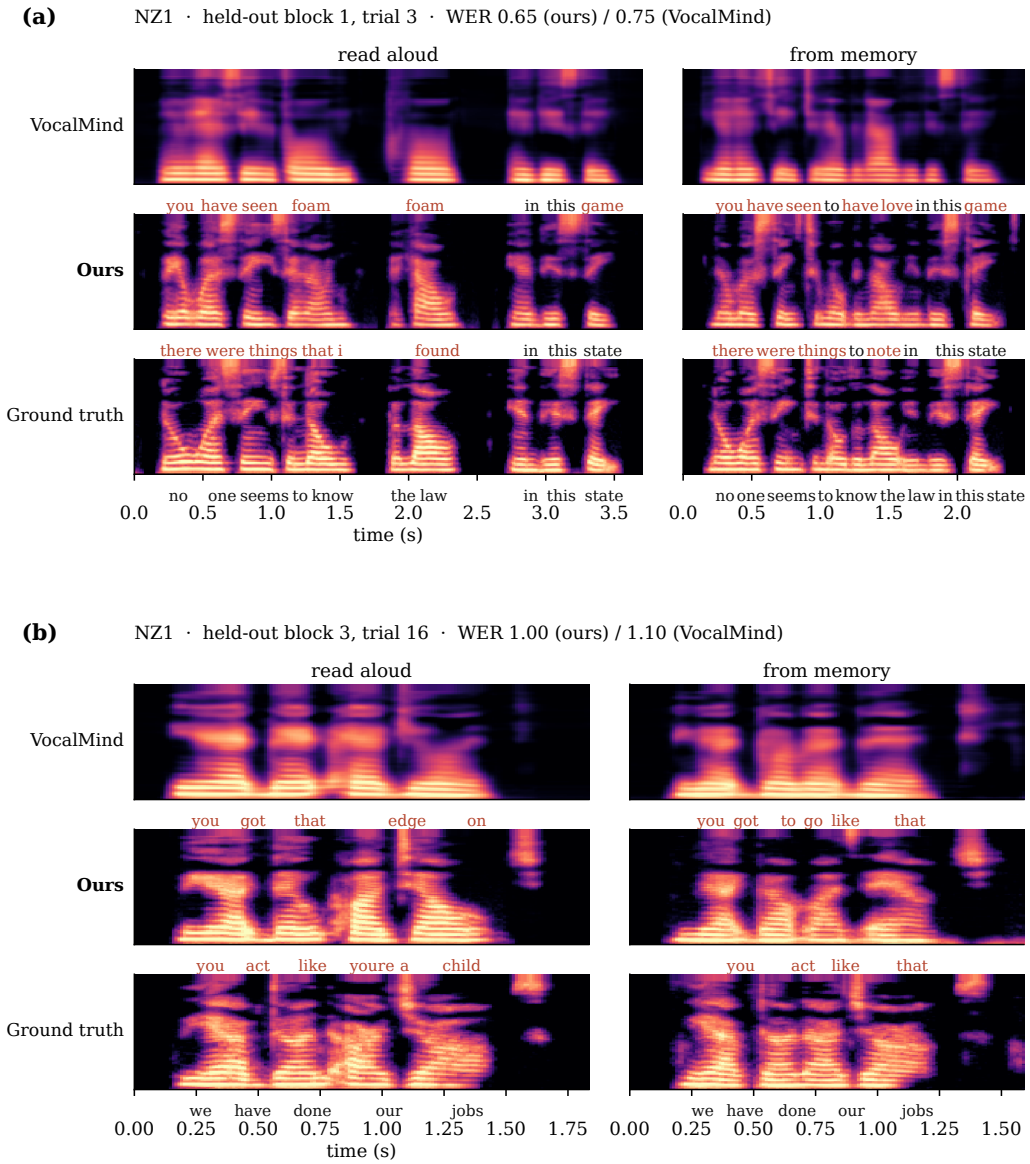


Figure 7: **Held-out trials decoded by the strongest published baseline and by our method on NZ1.** Description of results follow from the previous figure.

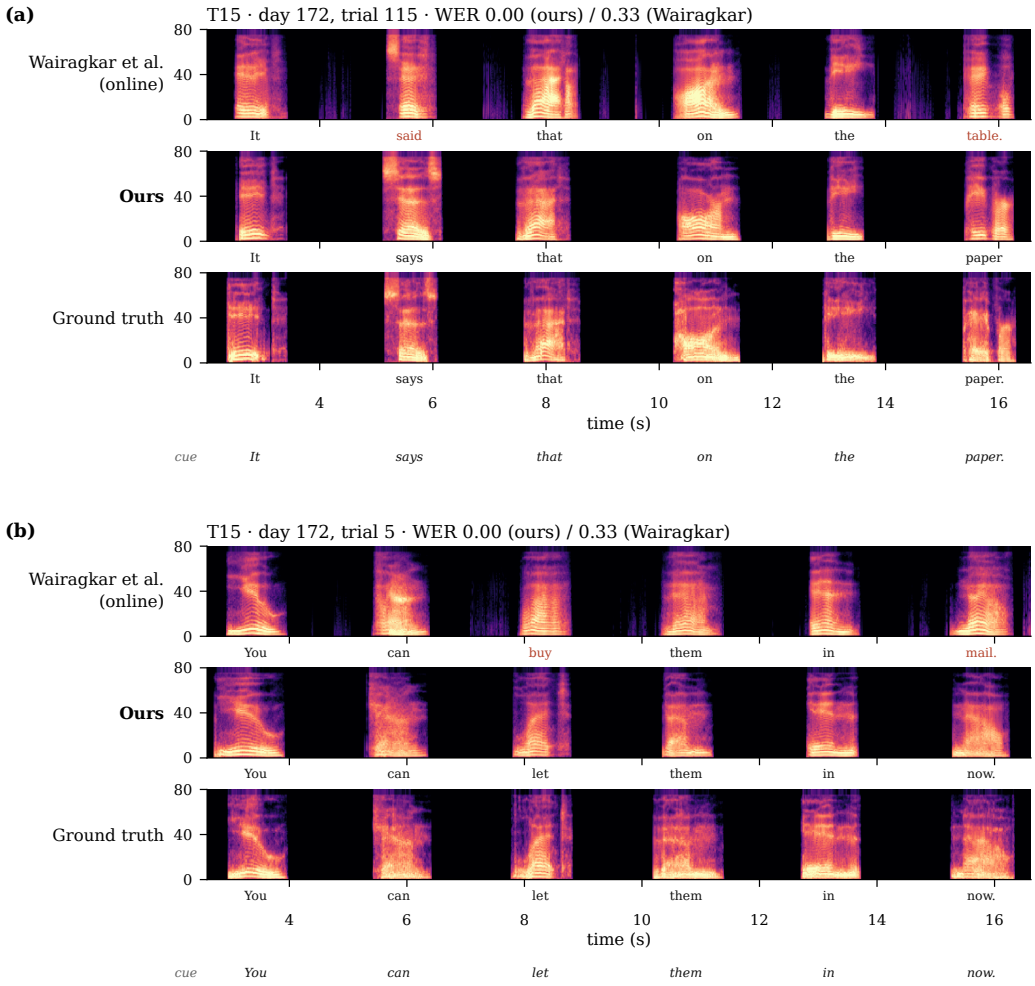


Figure 8: **Held-out trials from test day 172 decoded by Wairagkar et al. and our method on T15.** Description of results follow from the previous figure.