Brain2voice 2.0: High-performance voice synthesis brain-computer interface

Maitreyee Wairagkar[1,†], **Aparna Srinivasan**[1,2], **Nicholas S. Card**[1], **Tyler Singer-Clark**[1,2], **Xianda Hou**[1,3], **Carrina Iacobacci**[1], **Lee M. Miller**[4,5,6], **Leigh R. Hochberg**[7,8,9], **David M. Brandman**[1,*,†], **Sergey D. Stavisky**[1,*,†]

[1]Department of Neurological Surgery, University of California, Davis
[2]Department of Biomedical Engineering, University of California, Davis
[3]Department of Computer Science, University of California, Davis
[4]Center for Mind and Brain, University of California, Davis
[5]Department of Neurobiology, Physiology and Behavior, University of California, Davis
[6]Department of Otolaryngology, Head and Neck Surgery, University of California, Davis
[7]School of Engineering and Carney Institute for Brain Sciences, Brown University
[8]VA RR&D Center for Neurorestoration and Neurotechnology, VA Providence Healthcare
[9]Center for Neurotechnology and Neurorecovery, Department of Neurology, Massachusetts General Hospital, Harvard Medical School

[†]Correspondence: mwairagkar@ucdavis.edu
dmbrandman@ucdavis.edu
sstavisky@ucdavis.edu

[*]Co-senior authors

Abstract

Brain-computer interfaces (BCIs) offer a promising solution to speech loss due to neurological injury by decoding intended speech directly from brain activity. While recent BCIs have restored high-accuracy text-based communication, they fail to provide instantaneous voice output essential for the natural flow of conversation. Brain-to-voice BCIs address this gap by decoding voice directly from neural signals. However, even the state-of-the-art (SOTA) BCI-synthesized voice is not yet intelligible enough for real-world adoption. We introduce brain2voice 2.0, a new multimodal Transformer-based BCI decoder architecture capable of synthesizing highly intelligible voice from intracortical neural signals in real-time. Brain2voice 2.0 is trained on continuous and custom-tokenized acoustic targets and phoneme targets, leveraging their complementary speech information. We use self-supervised and adversarial training objectives that enhance acoustic feature quality and improve synthesis intelligibility. At each 10 ms timestep, the model causally outputs continuous and tokenized acoustic features for real-time voice synthesis as well as time-aligned phoneme predictions (raw phoneme error rate: 7%, comparable to the latest brain-to-text models). We evaluated this new approach on our prior intracortical brain-to-voice benchmark dataset (Wairagkar et al. 2025). Naive human listeners transcribed brain2voice 2.0 synthesized voice with a word error rate of 5.24%—an 8$\times$ improvement in intelligibility over previous SOTA results (43.75%). Brain2voice 2.0 demonstrates that highly intelligible real-time voice synthesis from neural signals is achievable, for the first time crossing the intelligibility threshold necessary for clinically viable brain-to-voice BCIs for people with paralysis.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

1 Introduction

Paralysis caused by neurological injury can rob people of the ability to speak. This severely impacts their independence and agency. Restoring naturalistic speech is a top priority for people with speech loss [1]. In recent years, brain-computer interfaces (BCIs) have made significant strides towards restoring communication by decoding intended speech directly from brain activity into text ("brain-to-text") or voice ("brain-to-voice"), essentially bypassing the neural injury [2–6].

Brain-to-text BCIs have demonstrated reliable high-accuracy speech decoding in multiple clinical trial participants [2–4, 7–9] and are being used for daily communication [10]. Other text-BCIs have decoded handwriting [11–15] and typing [16–18]. Despite restoring essential communication, text-based BCIs do not fulfill real-time conversation needs. While text-BCIs provide some real-time feedback by displaying words or characters on the screen as they are decoded, the decoded sentences are only finalized by a separate language model at the end of the utterance, at which point it can be transmitted as text or played aloud by a text-to-speech (TTS) software [10, 19]. The shortfall of this delayed communication becomes prominent when, for instance, talking to a BCI user without looking at their screen (e.g., over the phone or across the room): the conversation can go silent for long periods while the BCI user is composing their message (including speech decoding and manual post-hoc corrections), leading to interruptions and misunderstandings. Communication with immediacy is essential for continuous conversation, which involves rapid turn-taking and interrupting each other—neither of which is possible with current text-based BCIs.

Brain-to-voice BCIs can synthesize voice instantaneously to enable the natural flow of speech. Our recent SOTA brain-to-voice study [5] demonstrated that voice can be decoded directly from neural activity and synthesized by a BCI with real-time audio feedback at 25 ms latency. This enables a person with ALS not only to speak in real-time, but also to modulate the pitch and intonation of their BCI-voice—an important component of human speech. Despite this progress, the synthesized voice was not consistently intelligible: human listeners understood the BCI-voice with a word error rate of 43.75%, which is insufficient for daily communication. Direct high-fidelity intelligible voice synthesis is challenging, as it requires high-resolution neural information to capture rich and variable speech features. Due to this demand, prior BCIs using lower-resolution neural recording modalities have yielded lower intelligibility [3, 20–22]; delivering high intelligibility neurally-synthesized voice is a critical need before brain-to-voice neuroprostheses are ready for real-world adoption.

Here, we present a new speech BCI architecture, 'brain2voice 2.0', to synthesize intelligible voice from high-resolution intracortical neural signals[1]. This model yielded a word error rate of 5.24% with human listener transcription on our prior SOTA brain-to-voice benchmark dataset [23], substantially improving the intelligibility of BCI-synthesized voice without compromising near-instantaneous latency[2]. Brain2voice 2.0 is a multimodal Transformer-based model trained on acoustic targets—both continuous (rapidly varying) and tokenized (structured)—and phonemic targets with self-supervised learning and adversarial training objectives. This leverages complementary information from multiple target sources—all of which have human brain activity correlates—to enhance acoustic feature quality and improve voice synthesis intelligibility. The multimodal brain2voice 2.0 simultaneously outputs (1) continuous acoustic features (LPCNet features), (2) tokenized acoustic features (custom LPCNet tokens), and (3) phonemes, causally at every 10 ms timestep. The predicted phonemes have a low causal raw phoneme error rate of 7% (without a language model), comparable to recent brain-to-text models [2, 4]. Importantly, these phonemes are automatically time-aligned with exact temporal precision to attempted speech—a novel capability that could further serve as a basis for synchronous text decoding. The predicted continuous and tokenized acoustic features are both converted to audio waveform using a pretrained autoregressive LPCNet vocoder [24], resulting in intelligible real-time voice synthesis. Brain2voice 2.0 demonstrates that high intelligibility is achievable by decoding intracortical neural signals, establishing a strong foundation for future closed-loop brain-to-voice BCIs.

[1] Code is available at `https://github.com/neuroprosthetics-lab/brain2voice2.0`.

[2] Examples of video and audio of brain2voice 2.0 synthesis are available at `https://neuroprosthetics-lab.github.io/brain2voice2.0`.

2 Related work

Brain-to-voice decoding remains relatively underexplored due to several challenges: the absence of ground-truth speech in people with speech loss, constrained signal-to-noise ratio of speech-relevant neural signals, limited training data, and the small number of chronic clinical trial participants with intracortical electrodes. To our knowledge, our prior work [5] is the only intracortical BCI study to demonstrate real-time continuous voice synthesis and has achieved the highest performance to date. As intracortical BCIs have consistently outperformed other neural recording modalities across different motor tasks [2, 4, 5, 11, 18], we focus on intracortical data to develop brain2voice 2.0.

Prior brain-to-voice studies using electrocorticography (ECoG) and stereoelectroencephalography (sEEG) are also limited, and the majority involve opportunistic brief studies with healthy speakers undergoing epilepsy monitoring, conducted offline and acausally. Only a small number of chronic BCI studies have been conducted with people with speech loss, which is our target population.

Among studies with healthy speakers, [25] was the first to synthesize continuous speech offline from ECoG. Subsequent studies [22, 26–30] synthesized single words, while [31–33] synthesized sentences, via direct decoding of acoustic features. However, the synthesized speech was largely unintelligible, with correlations between synthesized and target speech ranging from 0.45 to 0.80, and no WER was reported. [34] used a discrete neural latent space for single-word decoding, but the approach was not compatible with causal real-time synthesis. [35] proposed two parallel pipelines for acoustic features and text, fused via TTS and voice cloning, reporting a WER of 18.9% in offline acausal decoding. Despite the advantage of having ground-truth speech from multiple participants, none of these studies achieved intelligible voice synthesis suitable for practical use.

Among chronic BCI studies with clinical trial participants with speech loss, [21] synthesized a six-word vocabulary online with 80% accuracy from a person with ALS who retained overt speech ability, using 128-channel ECoG, but with a delay of 2 s after each word. [3] synthesized speech acausally at the end of each sentence from an individual with vocal paralysis using 253-channel ECoG, achieving a WER of 54.4%. [20] demonstrated closed-loop synthesis from 253-channel ECoG in a participant with anarthria, but with a synthesis latency of ~1.7 s—exceeding the threshold for natural conversational feedback—and a WER of 58.8%. These studies fall short of the accuracy and latency requirements for practically usable voice synthesis, highlighting the need for a high-intelligibility, real-time brain-to-voice BCI system.

3 Methods

3.1 Participant and intracortical neural dataset

We use our prior dataset, the most extensive publicly available intracortical neural dataset [23], on which we demonstrated previous SOTA brain-to-voice performance [5]. This dataset was recorded from BrainGate2 clinical trial (ClinicalTrials.gov Identifier: NCT00912041) participant 'T15', a 45-year-old man with amyotrophic lateral sclerosis (ALS) and severe dysarthria over a 195-day period. This pilot clinical trial was approved under an Investigational Device Exemption by the US Food and Drug Administration (#G090003). Permission was also granted by the Institutional Review Boards at the University of California, Davis (#1843264 and #1950310) and Mass General Brigham (#2009P000505).

Neural activity was recorded from four microelectrode arrays (256 total electrodes) placed in the ventral precentral gyrus (Fig. 1) as T15 attempted to speak sentences cued on screen (8,489 unique trials). The dataset contains denoised threshold crossings (-4.5×RMS threshold) and spike-band power (250-5000 Hz) binned into 10 ms bins, normalized with a 10 s rolling window, causally smoothed, and log-transformed, yielding 512 neural features at 10 ms timesteps. Further signal processing and feature extraction details are described in [5]. Additionally, the dataset provides speech segmentation derived from the brain-to-voice model (onset and offset of each attempted word) and synthetic acoustic (LPCNet) features time-aligned with each neural trial.

3.2 Brain2voice 2.0 model architecture

Brain2voice 2.0 is a multimodal Transformer-based model [36] that causally decodes continuous acoustic features, custom acoustic tokens, and phonemes at each 10 ms timestep from an input

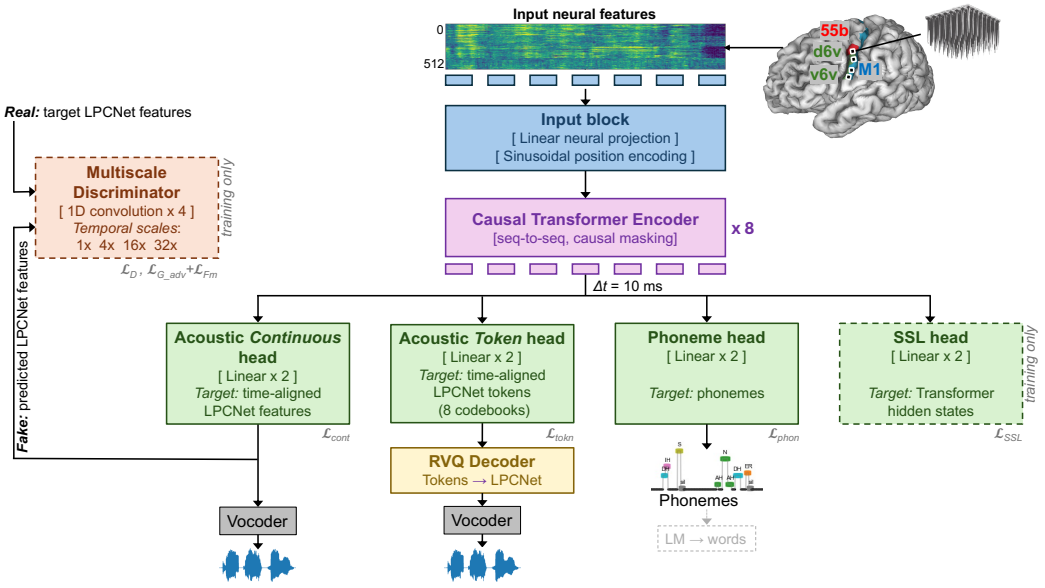


Figure 1: **Brain2voice 2.0 neural decoder for instantaneous voice synthesis.** A multimodal causal Transformer receives intracortical neural input from four microelectrode arrays in speech motor cortex of T15: ventral precentral gyrus (v6v and d6v), primary motor cortex (M1), and middle precentral gyrus (55b), of participant T15 with ALS and outputs at each 10 ms timestep: (1) continuous LPCNet features, (2) LPCNet tokens, (3) phoneme probabilities, and (4) Transformer hidden state for SSL. A multiscale discriminator provides adversarial training.

sequence of neural features (Fig. 1). The input neural features ($t \times 512$) are first passed through a linear projection layer followed by sinusoidal positional encoding. This projected sequence is then passed to the model backbone—a multilayer (8 layers) sequence-to-sequence Transformer encoder with causal masked self-attention (hidden size: 384, 6 heads). The resulting hidden states are passed to four parallel output heads: (1) a continuous acoustic head, (2) a tokenized acoustic head, (3) a phoneme head, and (4) a self-supervised learning (SSL) head, enabling multimodal predictions at each timestep. Additionally, a multiscale discriminator is used for adversarial training of the continuous acoustic outputs. All model hyperparameters are listed in Appendix Tables A1 and A2.

Continuous acoustic head: The continuous acoustic head predicts 20-dimensional continuous LPCNet features—the primary output for direct naturalistic voice synthesis. It consists of two linear layers, with rectified linear unit (ReLU) activation after the first layer. At each timestep, the predicted feature vector is converted to a 10 ms voice frame by the pretrained LPCNet vocoder [24]. Training targets are LPCNet features extracted from time-aligned synthetic speech (see Model training section).

Tokenized acoustic head: The tokenized acoustic head predicts LPCNet token probabilities across multiple codebooks trained using a custom tokenizer via parallel output heads, each with two linear layers with ReLU activation after the first. At each timestep, each head independently predicts a discrete token via classification. The LPCNet feature vector is reconstructed from the predicted tokens and converted to a 10 ms voice frame by the vocoder. Custom tokenizer is described below.

Phoneme head: The phoneme head predicts probabilities over 39 phonemes, a silence token, plus a blank token at each timestep via classification. It consists of two linear layers with ReLU activation after the first. The training target is the phoneme sequence derived from the trial text cue, providing the model with coarser-level, linguistically grounded supervision.

Self-supervised learning head: The SSL head predicts Transformer hidden states from masked neural inputs [37], providing auxiliary supervision beyond the acoustic and phonemic targets. Random spans of neural inputs are masked before the forward pass, and the SSL head, consisting of two linear layers with ReLU after the first, reconstructs the hidden states at masked positions, regularizing the Transformer and improving generalization.

Multiscale discriminator: We use adversarial training with a multiscale discriminator to enhance acoustic feature quality. The discriminator operates on the continuous LPCNet features at four temporal scales: full resolution (10 ms, frame-level), 4× (40 ms, sub-phonemic), 16× (160 ms, syllabic), and 32× (320 ms, word-level), via average pooling. Each scale uses an identical four-layer 1D convolutional network with progressively expanding channels (20, 64, 128, 256), capturing local temporal patterns at increasing levels of abstraction. It produces a frame-wise realness score, with intermediate feature maps. This multiscale design promotes acoustically realistic predictions across frame, phonemic, syllabic, and word-level temporal structures critical for perceptual speech quality.

3.3 Custom acoustic tokenizer using residual vector quantization

This custom tokenizer quantizes continuous LPCNet features into discrete tokens, restricting the output space to a fixed tokens derived by learning representative codebooks, providing complementary information to the continuous acoustic targets. Crucially, tokenizing time-aligned LPCNet features preserves speaker-specific properties—cadence, voice characteristics, and speech patterns such as slow disarthric speech—which general-purpose tokenizers like HuBERT [38] trained on healthy speech cannot capture. These tokenizers also cannot directly decode target voice, require separate vocoder training, and do not preserve the temporal precision or latency essential for real-time closed-loop synthesis. Our custom tokenizer, by contrast, operates entirely in LPCNet feature space without latent embeddings (unlike EnCodec [39] or DAC [40]), is fully reversible and interpretable, and preserves exact temporal alignment between predicted tokens, attempted speech, and vocoder output.

The tokenizer is built using residual vector quantization (RVQ) with MiniBatch K-means clustering on LPCNet features. It consists of 8 sequential codebooks, each with 128 centroids, trained on 20-dimensional LPCNet features. Each codebook is fit sequentially to the residual after subtracting the quantized reconstruction of all previous codebooks (Fig. 2), progressively refining quantization error. The tokenizer was trained exclusively on target LPCNet features of training trials from [23], with evaluation trials held out. The tokenizer can be retrained iteratively on new data, adapting to new vocabulary or speaking style. During *encoding*, each LPCNet vector is mapped to 8 token indices—one per codebook—by sequentially assigning each residual to its nearest centroid and subtracting the corresponding quantized value. During *decoding*, the 8 token indices are mapped back to their respective centroids and summed across all codebooks to reconstruct the LPCNet features.

3.4 Model training

3.4.1 Target data generation:

Brain2voice 2.0 is trained using three time-aligned supervised targets at 10 ms resolution: continuous LPCNet acoustic features, tokenized LPCNet acoustic features, and phoneme sequences.

Continuous acoustic target generation: The dataset [23] provides time-aligned synthetic LPCNet features as a proxy for T15's intended speech. These are generated by time-stretching TTS audio of text cue to match T15's word-level onset and offset timings—estimated from microphone recordings in the first session and from the brain-to-voice model in subsequent sessions. LPCNet features (18 spectral features, pitch period, and pitch strength) were extracted from this time-aligned audio. Here, we improved upon this pipeline by extracting LPCNet features from high quality own-voice TTS audio by cloning T15's pre-ALS voice samples first, and time-stretching in the LPCNet feature domain via cubic interpolation, rather than stretching the raw waveform [5]. This reduces time-stretching artifacts and yields higher-quality acoustic targets.

Tokenized acoustic target generation: We trained the custom RVQ tokenizer on all time-aligned continuous LPCNet features from the training trials. Each continuous LPCNet feature sequence was then encoded using the trained tokenizer to obtain 8 discrete tokens (one per codebook) per 10 ms timestep, which were used as targets for the tokenized acoustic head.

Phoneme target generation: Phoneme sequences were obtained from trial text cues using a grapheme-to-phoneme library [41] and used directly as targets for the phoneme head. Unlike the acoustic targets, precise temporal alignment for phonemes is not required, as the phoneme head is trained using connectionist temporal classification (CTC) loss [42], handling alignment implicitly.

These three complementary targets aligned to the neural data provide the model with continuous acoustic detail, discretized acoustic structure, and high-level phonemic guidance.

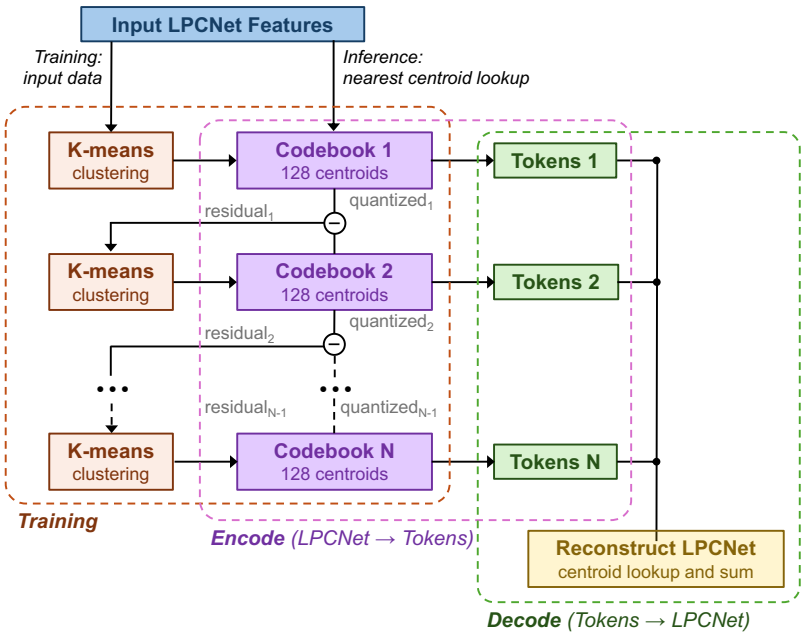


Figure 2: **Custom LPCNet tokenizer using residual vector quantization (RVQ).** The RVQ tokenizer discretizes continuous LPCNet features by learning 8 K-means codebooks sequentially by fitting residual of the previous stage. Encoding maps features to token indices via nearest-centroid lookup on successive residuals. Decoding reconstructs features by summing corresponding centroids.

3.4.2 Loss functions

We use multiple primary and auxiliary losses for different output heads, that together supervise the model using complementary speech representations with distinct neural correlates:

Continuous acoustic loss ($\mathcal{L}_{\text{cont}}$): Mean squared error loss is used for continuous acoustic feature regression in unrestricted continuous output space, with upweighting for harder-to-predict high-frequency features. Since the participant's dysarthric speech contains long silences that dilute the loss, speech timesteps are upweighted using a mask derived from target LPCNet features. The speech-to-silence weight ratio is linearly increased from 1:1 to 5:0.2 over the first 30% of training.

Tokenized acoustic loss ($\mathcal{L}_{\text{tokn}}$): Cross-entropy loss is used for classifying discrete tokens for each codebook at each timestep, combined as a harmonically weighted sum—reflecting the decreasing importance of later codebooks that progressively refine smaller residuals.

Phonemic auxiliary loss ($\mathcal{L}_{\text{phon}}$): CTC loss is used for predicting phoneme logits, which does not require time-aligned phoneme targets with neural input unlike acoustic targets.

Adversarial training auxiliary losses: Three adversarial losses are computed on the continuous acoustic features after a warm-up period: (1) discriminator loss ($\mathcal{L}_{\text{D}}$) trains the discriminator to distinguish real from predicted LPCNet features using softplus loss [43] averaged across four scales; (2) generator loss ($\mathcal{L}_{\text{G_adv}}$) maximizes the discriminator score on predicted outputs; (3) feature matching loss ($\mathcal{L}_{\text{Fm}}$) minimizes mean $L1$ distance between discriminator intermediate feature maps of real and predicted outputs for perceptually realistic predictions.

Self-supervised auxiliary loss ($\mathcal{L}_{\text{SSL}}$): The SSL loss minimizes cosine dissimilarity between predicted and target Transformer hidden states at masked positions [37], helping the encoder learn robust representations. Targets are obtained from an unmasked forward pass.

Finally, the primary continuous and tokenized acoustic losses are combined using learned uncertainty weighting [44] because these two losses differ by an order of magnitude. Each loss is scaled by a learnable weight derived from its log variance, automatically down-weighting higher-uncertainty losses without manual tuning. The total training loss is:

$$\mathcal{L}_{\text{total}} = w_1\mathcal{L}_{\text{cont}} + w_2\mathcal{L}_{\text{tokn}} + w_3\mathcal{L}_{\text{phon}} + (w_4\mathcal{L}_{\text{G_adv}} + w_5\mathcal{L}_{\text{Fm}}) + w_6\mathcal{L}_{\text{SSL}} \quad (1)$$

Where, w_1 and w_2 are tuned automatically, w_{3-6} are fixed (tuned manually on validation data): $w_3 = 0.1$, w_4 is linearly increased from 0.01 to 0.1 over 10,000 steps after adversarial training warm-up period, $w_5 = 2.0$, and $w_6 = 0.3$.

3.4.3 Model training details

The model was trained using AdamW optimizer (lr = 1e-3, weight decay = 0.01, fused) with cosine decay and linear warm-up over 25,000 batches. Learnable loss weights for acoustic features were jointly optimized at lr = 1e-3 without weight decay. The discriminator used a separate AdamW optimizer (lr = 5e-5, weight decay = 0.01, non-fused) updated every alternate batch after a 5,000-step warm-up to stabilize adversarial training. Gradient clipping (max norm 1.0) and bfloat16 mixed precision were applied throughout. All components were compiled for faster training. Batches of 32 training trials each were randomly sampled and a random contiguous window equal to the context length of the model (6.4 s) was extracted from each trial. Batches were downsampled from 10 ms to 50 ms resolution (5 bins) for compute efficiency. Neural features were augmented during the last 20% of training with Gaussian noise, baseline offsets, cumulative drift noise, and 10% channel dropout to improve generalization and give an additional boost near the end. Hyperparameters were tuned manually at each development stage. 128 benchmark trials from [23] were held out for for testing and comparison with prior SOTA [5], the remaining trials were split into 30 validation trials and the rest for training. All training was performed on a single NVIDIA RTX 5090 GPU and took approximately 1 hour to complete.

3.4.4 Simulated real-time inference

Brain2voice 2.0 is fully compatible with real-time closed-loop BCI decoding pipeline. We simulated real-time inference to evaluate model performance as it would operate in a closed-loop BCI. Neural input was decoded using a sliding window of 80 bins (800 ms) shifted by one 10 ms bin at each timestep—shorter than the training context (6.4 s) for faster inference. Only the last prediction from each window was retained, yielding new predictions at every 10 ms.

(1) Continuous acoustic output: Predicted continuous LPCNet features were rescaled and passed to the pretrained vocoder to synthesize each 10 ms audio frame. This was the primary evaluation output.

(2) Tokenized acoustic output: Predicted logits for each of the 8 codebooks were smoothed via exponential moving average, the most likely (argmax) token per codebook was selected and LPCNet features were reconstructed via the RVQ decoder, which were rescaled, causally smoothed, and passed to the vocoder at every 10 ms timestep.

(3) Phoneme output: The most likely (argmax) phoneme was selected from predicted logits at each timestep, blank tokens were removed, and consecutive repeated phonemes were collapsed.

The model operated entirely causally—all predictions at each 10 ms timestep used only past neural data, representing the most challenging inference scenario of instantaneous voice synthesis directly translatable to real-world closed-loop BCIs.

3.4.5 Evaluating intelligibility of synthesized speech

We evaluated the intelligibility of brain2voice 2.0 synthesized voice by recruiting human listeners via a crowdsourcing platform to transcribe held-out testing trials (benchmark and video sets from [23]). Each trial was independently transcribed by 7 listeners, and the median word error rate (WER) and corresponding phoneme error rate (PER) across transcriptions for each trial as in [5], were used as the primary intelligibility metrics (see A.5 for more details). We also transcribed voice samples through automatic speech recognition (ASR) via Whisper [45] to ensure consistency and scalability across conditions for comparative analyses. Additionally, we computed the Pearson correlation coefficient and mel-cepstral distortion (MCD) as objective acoustic quality metrics, following prior studies.

4 Results

4.1 Voice synthesis from intracortical neural signals

Quality of predicted acoustic features. High-quality voice was synthesized in real-time from continuous and tokenized LPCNet acoustic features predicted from intracortical neural data. At each 10 ms timestep, a new frame of LPCNet features was causally predicted. Fig. 3 shows the predicted continuous and tokenized LPCNet features alongside their target features, the synthesized voice waveform, and spectrograms. The predicted features closely matched the targets, with a Pearson correlation coefficient of $r = 0.77 \pm 0.06$ and mean squared error (MSE) of 0.018 ± 0.005 (mean ± s.d.) across all 20 normalized features (for tokenized features, $r = 0.68 \pm 0.07$ and MSE = 0.029 ± 0.007), including the harder to predict higher-frequency features that contribute to voice intelligibility.

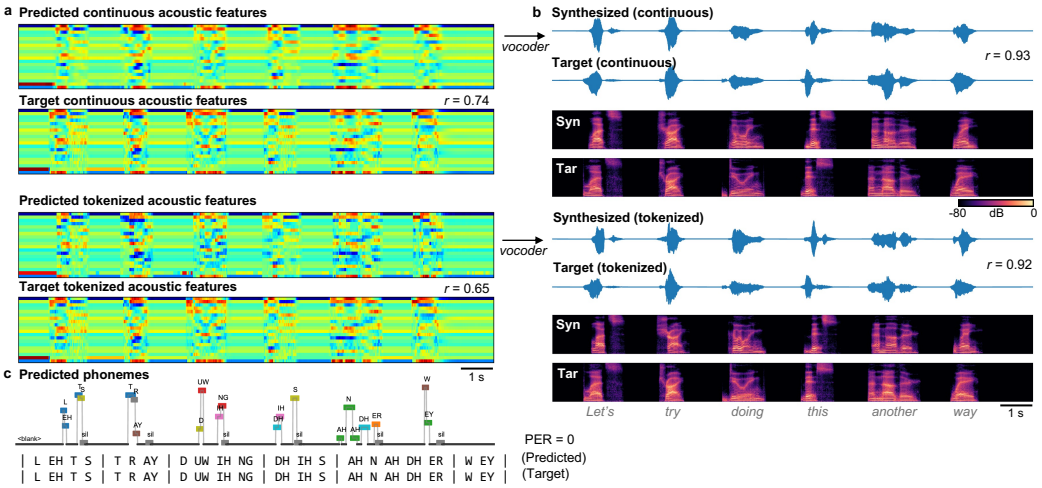


Figure 3: **Brain2voice 2.0 acoustic and phonemic feature prediction and voice synthesis. (a)** Predicted and target continuous ($r = 0.74$) and tokenized ($r = 0.65$) LPCNet features for an example trial. **(b)** Synthesized and target waveforms and spectrograms from continuous ($r = 0.93$) and tokenized ($r = 0.92$) outputs. **(c)** Predicted phoneme sequence (PER 0%) that is automatically time-aligned to the synthesized voice.

Intelligibility of synthesized voice. The intelligibility of brain2voice 2.0 synthesized voice as transcribed by naïve human listeners is shown in Table 1. The continuous acoustic output achieved an average WER of 5.24% and PER of 3.96%, and the tokenized acoustic output achieved WER of 5.65% and PER of 5.36% on the benchmark test trials. 79% and 74% of synthesized sentences were fully intelligible (0% WER), respectively, with a median WER and PER of 0%. Brain2voice 2.0 yielded an 8× improvement in intelligibility over the previous SOTA WER of 43.75% [5], establishing a new SOTA for brain-to-voice BCIs. Table 1 also shows intelligibility metrics on the video example set from [23] and ASR-based transcription error rates. A detailed comparison with the baseline model is provided in Appendix A.2. Example synthesized voice trials are provided alongside the target voice[3]. Appendix A.3 shows the ablation studies on the components of the brain2voice 2.0 model. Appendix A.4 shows the effect of electrode count and array location on the intelligibility of voice synthesis.

Objective acoustic quality of synthesized voice. Table 2 shows objective acoustic quality metrics for continuous and tokenized synthesized voice compared to their respective target voices on benchmark trials, along with the baseline comparison. For continuous voice, Pearson correlation across 40 Mel frequencies was $r = 0.93 \pm 0.02$ and MCD = 0.94 ± 0.22 dB (mean ± s.d.), with comparable results for tokenized voice ($r = 0.92 \pm 0.03$, MCD = 1.10 ± 0.25 dB). These results also show substantial improvements over the prior study [5] ($r = 0.83 \pm 0.04$, MCD = 2.89 ± 0.51 dB). All metrics were computed on speech periods only, with silences removed, to avoid inflating performance due to long

[3]Examples of video and audio of brain2voice 2.0 synthesized speech are available at `https://neuroprosthetics-lab.github.io/brain2voice2.0`.

Table 1: Intelligibility of Brain2voice 2.0 synthesized voice.

	Continuous Brain2voice 2.0			**Tokenized Brain2voice 2.0**		
	Mean error rate (%) [95% CI]		Error-free	Mean error rate (%) [95% CI]		Error-free
	WER ↓	**PER** ↓	trials (%) ↑	**WER** ↓	**PER** ↓	trials (%) ↑
Human transcription (benchmark test set)	**5.24** [2.89, 7.94]	**3.96** [2.25, 5.90]	**78.89**	5.65 [3.31, 8.43]	5.36 [3.23, 7.67]	74.44
Human transcription (video set)	2.11 [0.26, 4.47]	1.45 [0.26, 2.93]	89.47	3.21 [0.44, 7.42]	2.39 [0.49, 4.66]	86.84
ASR transcription (benchmark test set)	8.04 [5.17, 11.26]	5.70 [3.66, 8.02]	70.00	7.64 [4.84, 10.88]	5.88 [3.84, 8.23]	68.89
ASR transcription (video set)	3.84 [0.26, 9.91]	1.61 [0.30, 3.32]	89.47	5.31 [1.72, 9.98]	4.64 [1.66, 8.29]	78.95

For comparison, the previous SOTA brain-to-voice reported in Wairagkar et al. [5], which is our baseline, achieved a WER of 43.75%, a PER of 34.00%, and 7.78% error-free trials (human transcription, benchmark test set). The full baseline comparison across evaluation sets and transcription methods is shown in Table A3.

inter-word silences in T15's dysarthric speech. High performance across both acoustic metrics is consistent with the intelligibility results.

Table 2: Objective acoustic quality of Brain2voice 2.0 synthesis compared to target speech

	Continuous voice synthesis		**Tokenized** voice synthesis	
	Pearson *r* [-1, 1] ↑ better	**MCD** dB, ↓ better	**Pearson *r*** [-1, 1] ↑ better	**MCD** dB, ↓ better
Brain2voice 2.0 (benchmark test set)	0.93 ± 0.02	0.94 ± 0.22	0.92 ± 0.03	1.10 ± 0.25
Baseline *Wairagkar 2025* [5] (benchmark test set)	0.83 ± 0.04	3.1 ± 0.54	N/A	N/A
Brain2voice 2.0 (video set)	0.94 ± 0.02	0.86 ± 0.19	0.93 ± 0.02	1.01 ± 0.22
Baseline *Wairagkar 2025* (video set)	0.85 ± 0.05	2.83 ± 0.47	N/A	N/A

Values are mean ± s.d.

4.2 Phoneme prediction from brain2voice 2.0

Brain2voice 2.0 also emitted phonemes at each 10 ms timestep, causally decoded from neural data with high accuracy. The average causal raw phoneme error rate on benchmark trials was PER = 7.03% (95% CI: [5.74, 8.39]) and on the video set was PER = 6.23% (95% CI: [3.98, 8.91]) (median PERs of 5.26% and 4.26% respectively), comparable to latest brain-to-text BCIs [2, 4]. Importantly, the predicted phonemes were automatically time-aligned with the synthesized speech and consequently with the participant's attempted speech (Fig. 3c), without requiring explicit aligned phoneme targets. This alignment emerges automatically from the time-aligned acoustic targets used for training. This is in contrast to CTC-based brain-to-text, where phoneme alignment is only approximated by the CTC and is not precisely aligned to speech onset (this partly stems from the neural activity for each word, including preparatory activity occurring hundreds of milliseconds before speech onset).

4.3 Latency

Brain2voice 2.0 is suitable for real-time inference, with total computation time well under 10 ms per incoming 10 ms neural bin. Inference times were measured during simulated real-time inference on a single NVIDIA RTX 5090. Inference time for continuous acoustic LPCNet features, including feature rescaling, was 1.47 ± 0.11 ms (mean ± s.d.). Inference time for tokenized features was 2.11 ± 0.21

ms, which additionally included causal logit smoothing, argmax token selection across codebooks, LPCNet feature reconstruction with the RVQ decoder, and causal LPCNet smoothing. Vocoding a single LPCNet frame took a further 1.2 ms. This leaves sufficient time for other computations required for closed-loop synthesis such as neural feature extraction, binning, normalization, smoothing, and sliding window buffering—all of which take under 4 ms as reported in [5]. Brain2voice 2.0 is therefore fully compatible with real-time closed-loop voice synthesis neuroprosthesis at 10 ms latency.

5 Limitations

This work used intracortical data from a single participant. While validation across multiple participants would strengthen generalizability, single-participant datasets are standard in chronic intracortical BCI research, given the significant scaling challenges involved—for example, recent high-impact chronic BCI studies all had a single clinical trial participant [2–5, 10, 11, 20, 46]. We evaluated existing data with some of the highest electrode density and signal quality available in speech-relevant brain regions. To our knowledge, no other publicly available datasets with sufficient neural signal fidelity for voice synthesis currently exist. However, it remains to be seen how performance scales with newer high-electrode count neural interfaces currently entering pilot clinical trials.

Model performance was assessed through offline evaluation with real-time inference simulation, enabling direct comparison against prior work. Offline decoding omits the potential effects of auditory feedback-driven neural adaptation by the user (which could be helpful or challenging). Evaluation in a closed-loop BCI setting remains a key next step to characterize real-world performance.

The model required ground-truth speech timing annotations from at least one seed session for initialization. Extending this to participants without precise speech onset markers, such as those with complete speech loss, is an open challenge that future work will need to address.

Finally, the dataset was collected under a structured task paradigm in which the participant attempted to speak visually-cued sentences. It remains to be seen how performance translates to self-generated spontaneous speech, as required for real-world communication.

6 Discussion

Brain2voice 2.0 is the most accurate and fastest voice synthesis BCI reported to date, and the first to achieve highly intelligible voice synthesis, with a human listener WER of 5.24%—an 8$\times$ improvement upon previously reported SOTA intelligibility [5]. This brings instantaneous voice communication BCIs closer to real-world viability for people with paralysis.

The model operates fully causally at each 10 ms timestep, making it suitable for future real-time closed-loop deployment. This is in contradistinction with prior voice decoding systems using ECoG and sEEG, which were neither fully causal nor real-time [3, 20, 21]. The multimodal training combining complementary targets achieved comparable intelligibility for both continuous and tokenized acoustic outputs, demonstrating that both representations successfully captured the underlying neural speech patterns. The multiscale discriminator spanning frame, phoneme, syllable, and word-level timescales improved the sharpness and perceptual quality of synthesized voice, while self-supervised learning stabilized model training by providing an auxiliary regularization.

We introduce a custom residual vector quantization-based tokenizer for LPCNet features that provides a complementary training target. It operates entirely in LPCNet feature space, is interpretable and invertible, and preserves speaker-specific voice properties and speech cadence—unlike tokenizers that operate in latent embedding space and discard speaker identity [38, 39]. This makes it particularly suitable for personalized voice synthesis. Additionally, the model's causal time-aligned phoneme predictions could serve as a basis for simultaneous synchronous text decoding, with phoneme error rates comparable to recent brain-to-text BCIs [2, 4] despite using a fully causal architecture.

Brain2voice 2.0 arrives at a critical moment—as academic and industrial BCI programs worldwide accelerate towards clinical deployment for speech restoration. This work provides the first demonstration of high-intelligibility neural voice synthesis, offering a strong architectural foundation for next-generation voice neuroprostheses.

Acknowledgments and disclosures

We thank participant T15 and his family and care partners for their immense contributions to this research.

Funding. Support was provided by Department of Defense CDMRP ALS Pilot Clinical Trial Award (AL250417 and AL220043); Searle Scholars Program, and a Career Award at the Scientific Interface from the Burroughs Wellcome Fund to Stavisky; A.P. Giannini Postdoctoral Fellowship, and Burroughs Wellcome Fund Career Award at the Scientific Interface to Card; University of California Davis School of Medicine Cultivating Team Science Award; University of California Davis Clinical and Translational Science Center (CTSC UL1-TR001860); NIH-NIDCD (U01DC017844); VA RR&D (A2295-R).

Disclosures. M.W., N.S.C., L.M.M, D.M.B., and S.D.S. are inventors on intellectual property related to intracortical BCIs owned by the Regents of the University of California. N.S.C. worked as a Visiting Neuroengineer at Neuralink from September 2025 to February 2026. L.R.H. has research agreements through MGH Translational Research Center with Neuralink, Synchron, Axoft, Precision Neuro, and Reach Neuro for which LRH provides consultative input. Mass General Brigham (MGB) is convening the Implantable Brain-Computer Interface Collaborative Community (iBCI-CC); charitable gift agreements to MGB, including those received to date from Paradromics, Synchron, Precision Neuro, Neuralink, and Blackrock Neurotech, support the iBCI-CC, for which LRH provides effort. D.M.B. is a Consultant at Globus Medical Inc and Principal investigator for Connect-One clinical trial with Paradromics Inc. S.D.S is an advisor to Subsense and was a consultant to Neuralink.

Author contributions. M.W., S.D.S., and D.M.B. conceived the study. M.W. conceptualized and designed the Brain2voice 2.0 architecture and implemented the decoder, the training algorithms, and the full framework for real-time brain-to-voice synthesis. M.W. also led the experiments, developed and coded the target speech generation algorithm, generated the cloned voice for participant T15, performed the ASR transcriptions, analyzed all data, created the figures, and wrote the manuscript. M.W. and A.S. performed the human listener evaluations. M.W., N.S.C., T.S.-C., and X.H. coded the real-time neural data collection system and built the neuroprosthetic cart system. M.W., N.S.C., and C.I. collected the primary data for this study. L.M.M. contributed to the human listener evaluations. D.M.B. led the planning and execution of the surgical implant procedure. L.R.H. is the sponsor-investigator of the multisite clinical trial. D.M.B. was responsible for all clinical trial-related activities at UC Davis. S.D.S. and D.M.B. supervised all aspects of the project. All authors reviewed and helped edit the manuscript.

Code and data availability. Code for the Brain2voice 2.0 model and the custom RVQ LPCNet tokenizer is available at `https://github.com/neuroprosthetics-lab/brain2voice2.0`. The data required to train and evaluate the model are publicly available at `https://doi.org/10.5061/dryad.2280gb64f`.

Video and audio examples of brain2voice 2.0 synthesized speech (continuous and tokenized), drawn from the same trials that human listeners transcribed, are available alongside the target voice and ground-truth text at `https://neuroprosthetics-lab.github.io/brain2voice2.0`. These examples help put the reported word error rates into perspective.

References

[1] Sergey D. Stavisky. Restoring Speech Using Brain-Computer Interfaces. *Annual Review of Biomedical Engineering*, 27(1):29–54, 2025.

[2] 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.

[3] 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.

[4] 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.

[5] 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.

[6] Nicholas S Card and Maitreyee Wairagkar. A rapidly deployable high-performance speech neuroprosthesis. In *Brain-Computer Interface Research: A State-of-the-Art Summary 12*, pages 11–17. Springer, 2025.

[7] Erin M. Kunz, Benyamin Abramovich Krasa, Foram Kamdar, Donald T. Avansino, Nick Hahn, Seonghyun Yoon, Akansha Singh, Samuel R. Nason-Tomaszewski, Nicholas S. Card, Justin J. Jude, Brandon G. Jacques, Payton H. Bechefsky, Carrina Iacobacci, Leigh R. Hochberg, Daniel B. Rubin, Ziv M. Williams, David M. Brandman, Sergey D. Stavisky, Nicholas AuYong, Chethan Pandarinath, Shaul Druckmann, Jaimie M. Henderson, and Francis R. Willett. Inner speech in motor cortex and implications for speech neuroprostheses. *Cell*, 188(17):4658–4673.e17, 2025.

[8] Samuel R. Nason-Tomaszewski, Pranav I. Deevi, Qinwan Rabbani, Brandon G. Jacques, Anna L. Pritchard, Lahiru N. Wimalasena, Brice A. Richards, Brianna M. Karpowicz, Payton H. Bechefsky, Nicholas S. Card, Darrel R. Deo, Eun Young Choi, Leigh R. Hochberg, Sergey D. Stavisky, David M. Brandman, Nicholas AuYong, and Chethan Pandarinath. Restoring brain-to-text communication in a person with dysarthria from pontine stroke using an intracortical brain-computer interface. *bioRxiv*, 2026.

[9] Justin J. Jude, Stephanie Haro, Hadar Levi-Aharoni, Hiroaki Hashimoto, Alexander J. Acosta, Nicholas S. Card, Maitreyee Wairagkar, David M. Brandman, Sergey D. Stavisky, Ziv M. Williams, Sydney S. Cash, John D. Simeral, Leigh R. Hochberg, and Daniel B. Rubin. Decoding intended speech with an intracortical brain-computer interface in a person with long-standing anarthria and locked-in syndrome. *Cell Reports*, 45(4), 2026.

[10] Nicholas S. Card, Tyler Singer-Clark, Hamza Peracha, Carrina Iacobacci, Xianda Hou, Maitreyee Wairagkar, Zachery Fogg, Elena Offenberg, Leigh R. Hochberg, David M. Brandman, and Sergey D. Stavisky. Long-term independent use of an intracortical brain-computer interface for speech and cursor control. *Nature Medicine*, pages 1–7, 2026.

[11] Francis R. Willett, Donald T. Avansino, Leigh R. Hochberg, Jaimie M. Henderson, and Krishna V. Shenoy. High-performance brain-to-text communication via handwriting. *Nature*, 593 (7858):249–254, 2021.

[12] Chaofei Fan, Nick Hahn, Foram Kamdar, Donald Avansino, Guy H. Wilson, Leigh Hochberg, Krishna V. Shenoy, Jaimie M. Henderson, and Francis R. Willett. Plug-and-Play Stability for Intracortical Brain-Computer Interfaces: A One-Year Demonstration of Seamless Brain-to-Text Communication. *Advances in neural information processing systems*, 36:42258–42270, 2023.

[13] Yu Qi, Xinyun Zhu, Xinzhu Xiong, Xiaomeng Yang, Nai Ding, Hemmings Wu, Kedi Xu, Junming Zhu, Jianmin Zhang, and Yueming Wang. Human motor cortex encodes complex handwriting through a sequence of stable neural states. *Nature Human Behaviour*, 9(6):1260–1271, 2025.

[14] Sowmya Manojna Narasimha, Jingya Huang, Ram Dyuthi Sristi, Vikash Gilja, and Gal Mishne. Compositional decoding of neural activity enhances generalization in handwriting BCIs. *bioRxiv*, 2026.

[15] Alisa D. Levin, Donald T. Avansino, Foram B. Kamdar, Nicholas S. Card, Maitreyee Wairagkar, Brandon G. Jacques, Justin J. Jude, Carrina Iacobacci, Bayardo E. Lacayo, Payton H. Bechefsky, Samuel R. Nason-Tomaszewski, Darrel R. Deo, Leigh R. Hochberg, Daniel B. Rubin, Ziv M. Williams, David M. Brandman, Sergey D. Stavisky, Nicholas AuYong, Chethan Pandarinath, Scott W. Linderman, Jaimie M. Henderson, and Francis R. Willett. Cross-brain transfer of high-performance intracortical speech and handwriting BCIs. *bioRxiv*, 2026.

[16] Chethan Pandarinath, Paul Nuyujukian, Christine H Blabe, Brittany L Sorice, Jad Saab, Francis R Willett, Leigh R Hochberg, Krishna V Shenoy, and Jaimie M Henderson. High performance communication by people with paralysis using an intracortical brain-computer interface. *eLife*, 6:e18554, 2017.

[17] Nishal P. Shah, Matthew S. Willsey, Nick Hahn, Foram Kamdar, Donald T. Avansino, Chaofei Fan, Leigh R. Hochberg, Francis R. Willett, and Jaimie M. Henderson. A flexible intracortical brain-computer interface for typing using finger movements. *bioRxiv*, 2024.

[18] Justin J. Jude, Hadar Levi-Aharoni, Alexander J. Acosta, Shane B. Allcroft, Claire Nicolas, Bayardo E. Lacayo, Nicholas S. Card, Maitreyee Wairagkar, Alisa D. Levin, David M. Brandman, Sergey D. Stavisky, Francis R. Willett, Ziv M. Williams, John D. Simeral, Leigh R. Hochberg, and Daniel B. Rubin. Restoring rapid natural bimanual typing with a neuroprosthesis after paralysis. *Nature Neuroscience*, pages 1–10, 2026.

[19] Hamza Peracha, Carrina Iacobacci, Tyler Singer-Clark, Leigh R Hochberg, Sergey D. Stavisky, David M. Brandman, and Nicholas S Card. A Personalized and Adaptable User Interface for a Speech and Cursor Brain-Computer Interface. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems*, pages 1–12, 2026.

[20] Kaylo T. Littlejohn, Cheol Jun Cho, Jessie R. Liu, Alexander B. Silva, Bohan Yu, Vanessa R. Anderson, Cady M. Kurtz-Miott, Samantha Brosler, Anshul P. Kashyap, Irina P. Hallinan, Adit Shah, Adelyn Tu-Chan, Karunesh Ganguly, David A. Moses, Edward F. Chang, and Gopala K. Anumanchipalli. A streaming brain-to-voice neuroprosthesis to restore naturalistic communication. *Nature Neuroscience*, 28(4):902–912, 2025.

[21] Miguel Angrick, Shiyu Luo, Qinwan Rabbani, Daniel N. Candrea, Samyak Shah, Griffin W. Milsap, William S. Anderson, Chad R. Gordon, Kathryn R. Rosenblatt, Lora Clawson, Donna C. Tippett, Nicholas Maragakis, Francesco V. Tenore, Matthew S. Fifer, Hynek Hermansky, Nick F. Ramsey, and Nathan E. Crone. Online speech synthesis using a chronically implanted brain–computer interface in an individual with ALS. *Scientific Reports*, 14(1):9617, 2024.

[22] Miguel Angrick, Maarten C. Ottenhoff, Lorenz Diener, Darius Ivucic, Gabriel Ivucic, Sophocles Goulis, Jeremy Saal, Albert J. Colon, Louis Wagner, Dean J. Krusienski, Pieter L. Kubben, Tanja Schultz, and Christian Herff. Real-time synthesis of imagined speech processes from minimally invasive recordings of neural activity. *Communications Biology*, 4(1):1–10, 2021.

[23] Maitreyee Wairagkar, Nicholas Card, Tyler Singer-Clark, Xianda Hou, Carrina Iacobacci, Lee Miller, Leigh Hochberg, David Brandman, and Sergey Stavisky. Data for: An instantaneous voice synthesis neuroprosthesis [dataset]. *Dryad*, 2025. URL `https://doi.org/10.5061/dryad.2280gb64f`.

[24] 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)*, pages 5891–5895, 2019.

[25] Gopala K. Anumanchipalli, Josh Chartier, and Edward F. Chang. Speech synthesis from neural decoding of spoken sentences. *Nature*, 568(7753):493–498, 2019.

[26] Christian Herff, Lorenz Diener, Miguel Angrick, Emily Mugler, Matthew C. Tate, Matthew A. Goldrick, Dean J. Krusienski, Marc W. Slutzky, and Tanja Schultz. Generating Natural, Intelligible Speech From Brain Activity in Motor, Premotor, and Inferior Frontal Cortices. *Frontiers in Neuroscience*, 13, 2019.

[27] 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.

[28] Yan Liu, Zehao Zhao, Minpeng Xu, Haiqing Yu, Yanming Zhu, Jie Zhang, Linghao Bu, Xiaoluo Zhang, Junfeng Lu, Yuanning Li, Dong Ming, and Jinsong Wu. Decoding and synthesizing tonal language speech from brain activity. *Science Advances*, 9(23):eadh0478, 2023.

[29] Julia Berezutskaya, Zachary V. Freudenburg, Mariska J. Vansteensel, Erik J. Aarnoutse, Nick F. Ramsey, and Marcel A. J. van Gerven. Direct speech reconstruction from sensorimotor brain activity with optimized deep learning models. *Journal of Neural Engineering*, 20(5):056010, 2023.

[30] Xiaolong Wu, Scott Wellington, Zhichun Fu, and Dingguo Zhang. Speech decoding from stereo-electroencephalography (sEEG) signals using advanced deep learning methods. *Journal of Neural Engineering*, 21(3):036055, 2024.

[31] Gaël Le Godais, Philémon Roussel, Florent Bocquelet, Marc Aubert, Philippe Kahane, Stéphan Chabardès, and Blaise Yvert. Overt speech decoding from cortical activity: a comparison of different linear methods. *Frontiers in Human Neuroscience*, 17:1124065, 2023.

[32] 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. *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023.

[33] 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, 2024.

[34] Yao Jia, Qi Lian, Lei Wang, Yueming Wang, and Yu Qi. Learning discrete neural latent spaces for high-performance speech decoding. *Journal of Neural Engineering*, 22(6):066039, 2025.

[35] Jiawei Li, Chunxu Guo, Chao Zhang, Edward F Chang, and Yuanning Li. High-fidelity neural speech reconstruction through an efficient acoustic-linguistic dual-pathway framework. *eLife*, 14:RP109400, 2026.

[36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. *Advances in Neural Information Processing Systems*, 30, 2017.

[37] Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. data2vec: A General Framework for Self-supervised Learning in Speech, Vision and Language. In *Proceedings of the 39th International Conference on Machine Learning*, pages 1298–1312, 2022.

[38] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460, 2021.

[39] Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High Fidelity Neural Audio Compression. *Transactions on Machine Learning Research*, 2023.

[40] Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar. High-Fidelity Audio Compression with Improved RVQGAN. *Advances in Neural Information Processing Systems*, 36:27980–27993, 2023.

[41] K Park and J Kim. g2pE, 2019. URL `https://github.com/Kyubyong/g2p`.

[42] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376, 2006.

[43] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, pages 2672–2680, 2014.

[44] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7482–7491, 2018.

[45] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust Speech Recognition via Large-Scale Weak Supervision. *Proceedings of the 40th International Conference on Machine Learning, PMLR*, 202:28492–28518, 2023.

[46] David A. Moses, Sean L. Metzger, Jessie R. Liu, Gopala K. Anumanchipalli, Joseph G. Makin, Pengfei F. Sun, Josh Chartier, Maximilian E. Dougherty, Patricia M. Liu, Gary M. Abrams, Adelyn Tu-Chan, Karunesh Ganguly, and Edward F. Chang. Neuroprosthesis for Decoding Speech in a Paralyzed Person with Anarthria. *New England Journal of Medicine*, 385(3): 217–227, 2021.

A Appendix

A.1 Brain2voice 2.0 model hyperparameters

This appendix lists the full configuration of brain2voice 2.0 implementation. Table A1 gives the model architecture hyperparameters, and Table A2 gives the training parameters, including the optimizer, learning rate schedule, loss weights, and data augmentation settings. Hyperparameters were tuned on the validation set and held fixed across all experiments.

Table A1: Hyperparameters for Brain2voice 2.0 model

Parameter	Value	Description
Brain2voice 2.0		
Input neural features	512	Threshold crossings + spike-band power from 256 electrodes
Context length	128	Neural input sequence of 6.4 s at reduced resolution of 50 ms bins (avg. 5 x 10 ms bins)
Positional encoding	Sinusoidal	Fixed sinusoidal positional encoding
Model dimension	384	Transformer hidden dimension
Attention heads	6	Multi-head self-attention heads
Transformer layers	8	Stacked causal encoder blocks
FFN dimension	1536	Transformer feed forward network dimension
Dropout	0.1	Applied to residual connections, feed forward network and attention weights
Output heads	4	Continuous and Tokenized acoustic, Phoneme, SSL
Continuous acoustic head	Linear x 2	Two linear layers with ReLU after first layer
Continuous output size	20	LPCNet feature dimensions
Tokenized acoustic head (x8 parallel heads)	Linear x 2	Two linear layers with ReLU after first layer, per codebook
Token vocabulary size	128	Discrete tokens per codebook
RVQ codebooks	8	Number of residual quantization stages
Phoneme head	Linear x 2	Two linear layers with ReLU after first layer
Phoneme output size	40	39 phonemes + CTC blank token
SSL head	Linear x 2	Two linear layers with ReLU after first layer
SSL input masking probability	prob=0.065, span=10	SSL neural input masking probability and span
Total parameters	16.5 M	
Multiscale Discriminator		
Discriminator type	Multiscale (4 scales)	1x, 4x, 16x, 32x average pooling corresponding to 10 ms, 40 ms, 160 ms, 320 ms scales
Architecture	1D CNN x 4	Four 1D convolutional layers
Input channels	20	Continuous LPCNet feature dimensions
Discriminator channels	64	Base channel width. Channel progression: 64 → 128 → 256 → 1
Kernel size	7	Applied at all convolutional layers
Padding	3	Same padding to preserve temporal dimension
Total parameters	1.2 M	
Custom RVQ tokenizer		
Input LPCNet features	20	Continuous LPCNet features
Codebooks	8	Sequential residual quantization stages
Token vocabulary	128	K-means centroids per codebook

Table A2: Hyperparameters for training Brain2voice 2.0

Parameter	Value	Description
Brain2voice 2.0		
Batch size	32	Trials per training step
Total training batches	25,000	Total training iterations
Max learning rate	1×10^{-3}	Peak learning rate after warm-up
Min learning rate	1×10^{-5}	Final learning rate at end of training
LR schedule	Cosine decay	Linear warm-up for first 10% of training
Optimizer	AdamW	Fused implementation for training efficiency
Weight decay	0.01	Applied to model parameters
Gradient clipping	1.0	Max gradient norm
Mixed precision	bfloat16	Via torch.autocast
Data augmentation onset	80% of training	Gaussian noise, offset, cumulative drift, channel masking
Learning rate adjustable loss weighting	1×10^{-3}	Applied to adjustable loss weighting parameters
Weight decay adjustable loss weighting	0	Applied to adjustable loss weighting parameters
Discriminator training		
Discriminator optimizer	AdamW	Non-fused implementation
Learning rate	5×10^{-5}	Constant learning rate for discriminator
Weight decay	0.01	Applied to discriminator parameters
Discriminator warm-up	5,000 steps	Discriminator is trained after 5,000 batches to allow initial Generator training stabilization
Gradient clipping	1.0	Max gradient norm
Discriminator update frequency	Every 2 batches	Alternating discriminator and generator updates
Loss weights		
Continuous and token acoustic loss weight	auto	Loss weights for continuous and token heads are automatically adjusted by learned params
Phoneme loss weight	0.1	Fixed weight for phoneme CTC loss
Adversarial loss weight	0.01-0.1	Linear ramp over 10,000 steps post warm-up
Feature matching weight	2.0	Fixed weight on feature matching loss
SSL loss weight	0.3	Fixed weight on SSL loss
Data augmentation		
Gaussian noise	$\sigma = 0.1$	White noise added to all features at every timestep
Constant offset	$\sigma = 0.05$	Baseline shift applied uniformly across trial
Offset scaling factor	0.67	Threshold crossings scaled relative to spike power
Cumulative drift	$\sigma = 0.005$	Slow cumulative drift noise accumulated along time
Channel masking prob	0.10	10% of channels are zeroed out
Augmentation onset	80%	Augmentation applied only in the last 20% batches

A.2 Comparison with baselines

Table A3 compares brain2voice 2.0 with the baseline model from Wairagkar et al. [5] on the same evaluation sets, using both human and ASR transcription. Because the baseline model produced only continuous synthesis, tokenized results are reported for brain2voice 2.0 alone. Brain2voice 2.0 reduced WER substantially across all evaluation sets and transcription methods.

Table A3: Intelligibility of Brain2voice 2.0 synthesized voice compared to baseline [5]

		Continuous voice synthesis			**Tokenized** voice synthesis		
Transcription type	**Model**	Error rate (%) [95% CI]		Error-free	Error rate (%) [95% CI]		Error-free
(data set)		**WER** ↓	**PER** ↓	trials (%) ↑	**WER** ↓	**PER** ↓	trials (%) ↑
Human transcription (benchmark test set)	**Brain2voice 2.0**	**5.24** [2.89, 7.94]	**3.96** [2.25, 5.90]	**78.89**	**5.65** [3.31, 8.43]	**5.36** [3.23, 7.67]	**74.44**
Human transcription (benchmark test set)	Baseline *Wairagkar 2025*	43.75 [35.93, 53.36]	34.00 [26.12 39.03]	7.78	N/A	N/A	N/A
Human transcription (video set)	**Brain2voice 2.0**	2.11 [0.26, 4.47]	1.45 [0.26, 2.93]	89.47	3.21 [0.44, 7.42]	2.39 [0.49, 4.66]	86.84
Human transcription (video set)	Baseline *Wairagkar 2025*	37.50 [27.34, 44.42]	23.73 [19.57, 32.25]	15.78	N/A	N/A	N/A
ASR transcription (benchmark test set)	**Brain2voice 2.0**	8.04 [5.17, 11.26]	5.70 [3.66, 8.02]	70.00	7.64 [4.84, 10.88]	5.88 [3.84, 8.23]	68.89
Human transcription (benchmark test set)	Baseline *Wairagkar 2025*	51.82 [43.59, 60.34]	44.04 [37.71, 51.1]	12.22	N/A	N/A	N/A
ASR transcription (video set)	**Brain2voice 2.0**	3.84 [0.26, 9.91]	1.61 [0.30, 3.32]	89.47	5.31 [1.72, 9.98]	4.64 [1.66, 8.29]	78.95
ASR transcription (video set)	Baseline *Wairagkar 2025*	43.05 [30.98 56.32]	35.82 [25.7 46.2]	26.32	N/A	N/A	N/A

Comparison with standard time-series architectures

Additionally, to place our results in the context of standard time-series architectures, we trained three baseline decoders (Transformer, LSTM, and CNN) to predict continuous acoustic features directly from neural data. We used continuous acoustic features because the continuous head produced the most intelligible voice output in brain2voice 2.0. Each baseline consisted of a single decoder followed by a linear output head, without the tokenized, phoneme, or SSL heads or adversarial training. The Transformer baseline uses the same encoder as brain2voice 2.0.

All models shared the brain2voice 2.0 training pipeline, including the batching protocol, context window, optimizer, learning rate, augmentation, training duration, and other training hyperparameters, and all were trained for causal, real-time synthesis. All three decoders had 8 layers and a hidden dimension of 384. The LSTM was unidirectional to preserve causality. The CNN was a causal, dilated 1D convolutional network (kernel size 3, dilation doubling per layer from 1 to 128) with left padding only, giving a receptive field of 511 timesteps. Evaluation set was also identical for all models.

Table A4 compares the performance of these common architectures. Brain2voice 2.0 outperformed all three baselines, and the Transformer outperformed the LSTM and CNN. On listening, speech synthesized by the baselines, particularly the LSTM and CNN, also had substantially lower perceptual quality.

These baseline models outperform the prior SOTA results reported in [5] because they benefit from our enhanced training pipeline. The pipeline of [5] differed substantially from that of Brain2voice 2.0, including its optimizer, loss function, context window, day-specific embedding layer, sequence-to-one training objective, and augmentation scheme. Our goal here was to test how different decoders perform within our current framework. We hypothesize that the performance of [5] would have also degraded with a LSTM or CNN decoder, as prior work has shown that Transformers outperform

Table A4: Comparison with common baseline time-series model architectures

Model type	**Continuous** voice synthesis	
	WER ↓	**PER** ↓
Brain2voice 2.0	**8.04** [5.17, 11.26]	**5.70** [3.66, 8.02]
Transformer	13.27 [9.46, 17.2]	10.29 [7.26, 13.46]
LSTM	18.16 [13.67, 22.99]	14.33 [10.73, 18.23]
CNN	22.42 [17.31, 27.91]	18.67 [14.33, 23.1]

Intelligibility was evaluated via ASR transcription

recurrent decoders for speech decoding [10]. Therefore [5] remains the appropriate published SOTA baseline for comparing the results of this study, as no other published model [3, 20–22] achieves comparable real-time synthesis of continuous sentences.

A.3 Ablation studies on Brain2voice 2.0 model components

To understand how each component of brain2voice 2.0 contributes to intelligibility, we ran three complementary ablation studies. First, we performed a cumulative ablation, progressively adding components to a minimal model and retracing the development of brain2voice 2.0. Second, we performed a leave-one-out ablation, removing individual components from the full model to isolate each one's contribution. Third, we performed a single-head ablation, testing a simplified model with only the continuous acoustic head. Across all ablations, we held loss coefficients, learning rate, and other hyperparameters fixed at the values tuned for the full model, since retuning each configuration separately would amount to optimizing a distinct architecture each time.

Ablation 1: Cumulative ablation

Table A5 presents an ablation study evaluating the contribution of each model design component of brain2voice 2.0. Starting from a baseline model with only the continuous acoustic head trained on original voice targets from our dataset in [5, 23], we progressively added components and improved voice targets. To evaluate each component's contribution, we assessed the intelligibility of synthesized speech at each ablation via human listener transcription, where each synthesized trial (continuous and tokenized) from the benchmark set was independently transcribed by 5–7 listeners (less than 15% ablation trials were transcribed by 5 or 6 listeners, majority were transcribed by 7 listeners).

The results suggest that the auxiliary heads benefit from being trained jointly: adding the tokenized head alone increased error rates, possibly because all hyperparameters were tuned on the full model and this configuration may be suboptimal without the other components, but the subsequent addition of the phoneme head improved performance, with all three heads together yielding lower error rates. Adversarial training improved both the intelligibility and perceptual quality of continuous synthesis. Adding the SSL head helped stabilize the training and allowed the model to train longer without overfitting. Improving voice target quality by using high-quality own-voice TTS audio and reducing time-stretching artifacts through alignment in the LPCNet feature domain was equally important in improving synthesis quality. Overall, all components jointly contributed to Brain2voice 2.0, achieving the best performance through a combination of architectural choices, training improvements, and voice targets.

Ablation 2: Leave-one-out ablation

To isolate the contribution of each key component within the final model, we removed the tokenized head, the phoneme head, adversarial training, or the SSL head one at a time and kept everything else fixed (Table A6). For scalability, we measured intelligibility in this ablation and the next one with ASR transcription rather than human transcription. Their absolute WERs are therefore not directly comparable to Table A5.

Table A5: Ablation 1 - Cumulative ablation

	Model configuration	Continuous voice synthesis		Tokenized voice synthesis		Phoneme causal raw
		WER ↓	**PER ↓**	**WER ↓**	**PER ↓**	**PER ↓**
1	Cont *(original voice targets)*	10.94 [7.5, 14.8]	8.76 [6.09, 11.71]	N/A	N/A	N/A
2	Cont + Tokn *(original voice targets)*	23.17 [18.71, 27.97]	21.15 [17.42, 24.98]	21.50 [16.93, 26.39]	18.83 [15.29, 22.44]	N/A
3	Cont + Tokn + Phon *(original voice targets)*	13.68 [10.49, 17.06]	12.63 [9.59, 15.81]	15.92 [12.36, 19.85]	13.12 [10.35, 15.97]	9.65 [8.17, 11.18]
4	Cont + Tokn + Phon *(improved voice targets)*	11.12 [8.14, 14.29]	10.70 [7.92, 13.72]	14.29 [10.9, 17.74]	12.40 [9.67, 15.33]	13.30 [11.58, 15.05]
5	Cont + Tokn + Phon + GAN *(improved voice targets)*	8.61 [5.85, 11.56]	7.90 [5.56, 10.43]	13.08 [9.59, 16.89]	11.46 [8.39, 14.79]	13.83 [12.09, 15.61]
6	Cont + Tokn + Phon + GAN + SW *(improved voice targets)*	9.28 [6.42, 12.33]	8.76 [6.14, 11.63]	11.81 [8.55, 15.38]	10.34 [7.58, 13.33]	11.73 [10.16, 13.39]
7	Cont + Tokn + Phon + GAN + SW + SSL *(improved voice targets)*	8.07 [5.06, 11.57]	7.08 [4.73, 9.55]	9.20 [6.22, 12.42]	7.20 [4.98, 9.6]	8.63 [7.11, 10.2]
8	**Brain2voice 2.0** Cont + Tokn + Phon + GAN + SW + SSL *(new own voice targets)*	**5.24** [2.89, 7.94]	**3.96** [2.25, 5.9]	**5.65** [3.31, 8.43]	**5.36** [3.23, 7.67]	**7.03** [5.74, 8.39]

Intelligibility of synthesized speech was assessed via human listener transcription on the benchmark test set for each ablation (mean error rate (%) [95% CI]).
Cont: continuous acoustic head
Tokn: tokenized acoustic head
Phon: phoneme head
GAN: adversarial training with multiscale discriminator
SW: upweighting continuous acoustic loss in speech regions
SSL: self-supervised learning head
original voice targets: time-aligned target LPCNet features from the [5, 23] dataset
improved voice targets: target features time-aligned by stretching in LPCNet feature domain rather than stretching target speech waveforms, to reduce time-stretching artifacts and improve acoustic quality
new own voice targets: target LPCNet features from high quality own-voice TTS audio cloned from T15's pre-ALS voice samples and aligned by time-stretching in the LPCNet feature domain.

Removing any single component degraded the performance and increased the WER relative to the full model. Removing SSL had the largest effect, followed by the phoneme head, the tokenized head, and adversarial training.

This result also clarifies the effect of adding the tokenized head in the cumulative ablation above, which increased the WER intermittently. However, removing this component in isolation reduces intelligibility, meaning the tokenized head helps performance. The effect of removing single components is somewhat modest. This suggests that the components in brain2voice 2.0 operate together, contributing through interactions that are difficult to isolate individually rather than through large, independent effects, complementing the findings of the cumulative ablation.

Ablation 3: Simplified continuous head ablation (Brain2voice 2.0 Lite)

We tested a simplified version of the model that keeps only the continuous acoustic head, removing the tokenized, phoneme, and SSL heads. We then added adversarial training, speech upweighting, and own-voice targets in sequence (Table A7; WERs were obtained from ASR transcription). Each addition improved intelligibility, and WER fell from 13.27% to 7.40%. Adversarial training, speech upweighting, and high-quality voice targets therefore each help even in a single-output configuration. This reduced variant, which we call Brain2voice 2.0 Lite, is a viable option when the added compute of a multi-head, multimodal model is undesirable, for example when simultaneous real-time phoneme decoding and quantized voice synthesis are not required.

Together, the three ablation studies show that each component is useful. Each contributes meaningfully to the best-performing configuration, and removing any single component degrades it. Because

Table A6: Ablation 2 - Leave-one-component-out ablation

	Model configuration	**Continuous** voice synthesis		**Tokenized** voice synthesis		**Phoneme** causal raw
		WER ↓	**PER ↓**	**WER ↓**	**PER ↓**	**PER ↓**
1	**Brain2voice 2.0**	**8.04** [5.17, 11.26]	**5.70** [3.66, 8.02]	**7.64** [4.84, 10.88]	**5.88** [3.84, 8.23]	7.03 [5.74, 8.39]]
2	No Tokenized head	11.19 [7.92, 14.7]	9.08 [6.51, 11.89]	N/A	N/A	11.12 [9.56, 14.03]
3	No Phoneme head	11.78 [8.37, 15.64]	7.85 [5.44, 10.43]	11.90 [8.53, 15.63]	7.36 [5.01 10.09]	N/A
4	No GAN	9.86 [6.66, 13.39]	5.87 [3.81, 8.11]	9.51 [6.7, 12.6]	5.91 [4.02, 7.87]	**6.85** [5.54, 8.24]
5	No SSL	12.48 [9.25, 15.94]	8.14 [5.83, 10.65]	13.45 [9.73, 17.59]	8.58 [6.12, 11.23]	10.73 [9.22, 12.3]

Intelligibility was evaluated via ASR transcription

Table A7: Ablation 3 - Simplified single-head ablation (Brain2voice 2.0 Lite)

	Model configuration	**Continuous** voice synthesis	
		WER ↓	**PER ↓**
1	Cont *(original voice targets)*	13.27 [9.46, 17.2]	10.29 [7.26, 13.46]
2	Cont + GAN *(original voice targets)*	11.64 [8.26, 15.43]	8.54 [5.99, 11.35]
3	Cont + GAN + SW *(original voice targets)*	9.55 [6.35, 13.34]	7.47 [5.09, 10.03]
4	**(Brain2voice 2.0 Lite)** Cont + GAN + SW *(own voice targets)*	**7.40** [4.71, 10.45]	**4.99** [3.15, 7.06]

Intelligibility was evaluated via ASR transcription

the effect of a component depends on the other components present, their contributions are not independent or additive. Rather than a single fixed architecture, brain2voice 2.0 therefore provides a modular recipe for high-accuracy, real-time, multimodal voice synthesis from intracortical signals, with the full model achieving our best human-evaluated intelligibility.

A.4 Contribution of electrode count and array location on voice synthesis intelligibility

We studied the effect of electrode count and array location on voice synthesis intelligibility. First, we randomly selected 64, 128, 192, and 256 electrodes from all arrays combined and trained separate brain2voice 2.0 models on each subset, evaluating synthesis intelligibility on benchmark trials via ASR transcription word error rate. WER decreased monotonically with increasing electrode count, though the improvement was marginal between 192 and 256 electrodes. Next, we trained separate models on each individual array to assess their relative contributions. WER was lowest for v6v, indicating it carries the highest speech-related SNR, while d6v contributed the least, yielding nearly unintelligible synthesis, consistent with prior report [4]. M1 and 55b performed intermediately, contributing more than d6v but less than v6v. These results demonstrate that voice synthesis intelligibility depends on both the number of electrodes and the speech-related SNR of the recording site, and that high number of electrodes from most relevant cortical areas are necessary for intelligible synthesis.

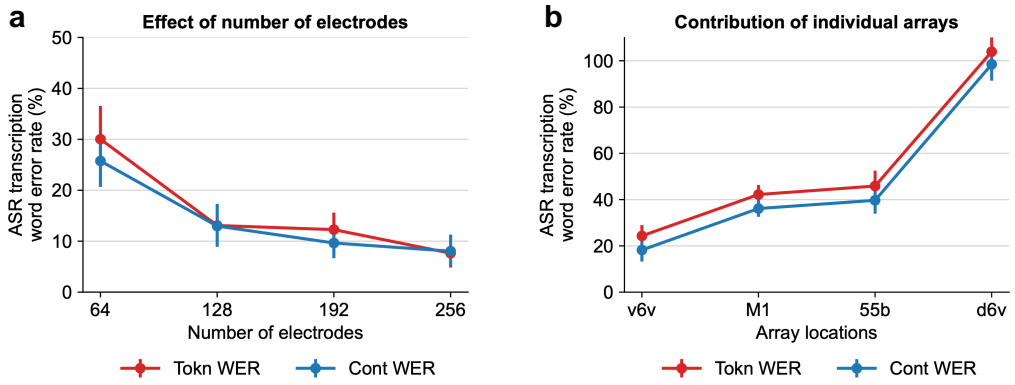


Figure A1: **Contribution of electrode count and array location to voice synthesis. (a)** WER on voice synthesis from models trained on randomly selected subsets of electrodes (error bars show 95% CI). WER decreases as number of electrodes increase, though the improvement becomes more marginal at higher counts. **(b)** Individual array contribution to voice synthesis measured by single-array WER. v6v contributed the most with the lowest WER, M1 and 55b contributed moderately, while d6v alone produced largely unintelligible synthesis. These results demonstrate that all four arrays together are necessary for achieving the highest possible voice synthesis quality.

A.5 Human listener transcription of BCI-synthesized voice for evaluating intelligibility

To evaluate the intelligibility of brain2voice 2.0, human listeners transcribed each BCI-synthesized trial from the benchmark set of [5]. To allow a direct comparison with the results reported in our previous study [5], we followed the same transcription protocol: each trial was transcribed independently by 7 different listeners, and the intelligibility score for each trial was taken as the median WER of these 7 transcriptions. Taking the median across multiple transcribers is a common way to combine WERs of the same sentence [3, 5, 20].

The listeners were recruited through a crowdsourcing platform, GoTranscript[4]. Each trial was posted as a separate job with custom instructions that any transcriber from the platform's pool, all proficient English speakers vetted by the platform, could accept and transcribe. In total, 67 unique transcribers participated, each transcribing between 1 and 132 trials (27 on average), compared to 28 transcribers who each transcribed 30–38 trials in [5]. However, both studies used an identical transcription protocol with 7 independent transcriptions per trial.

Because brain2voice [5] was originally evaluated by a different groups of transcribers, we performed two additional analyses to rule out evaluator differences as the source of improvement. First, we re-evaluated the synthesized trials from brain2voice [5] on the same crowdsourcing platform used for brain2voice 2.0. This yielded a WER of 37.5% on the same benchmark trials, consistent with the range reported in [5], indicating that transcription performance on the new platform is comparable to the previous evaluation. Second, to control for individual transcriber differences, we identified the transcribers who had transcribed both model versions. Twenty-three transcribers transcribed the same 128 sentences from both models, with brain2voice [5] always transcribed first. On these identical sentences, they had an average WER of 6.66% for brain2voice 2.0 versus 33.07% for brain2voice [5]. Together, these results show that the large improvement in intelligibility is not explained by transcriber-specific performance and reflects a genuine advance from our new approach.

Instructions for human listeners for transcribing BCI voice

We provided the following instructions to human listeners to transcribe the brain2voice 2.0-synthesized voice on benchmark trials:

Listen to the audio clip carefully as many times as needed and write down what you hear below each audio. Please use headphones to listen to the audio.

[4]Crowdsourcing platform used for human transcription `https://gotranscript.com/`.

- Use only valid English words from the English vocabulary. Do not use any digits or numeric characters. All numbers must be written out as words (e.g., “hundred people” not “100 people”).
- The audio may be unclear, and some words may be difficult to understand. Write down the word closest to what you hear. For example, if the word is “still” and you hear something like “cell”, write “cell”.
- If you do not understand a word at all, put a ? symbol in its place.
- Ensure that the number of words you transcribe (including ? symbols) is equal to the number of words you hear in the audio.
- The transcribed sentence does not have to be grammatically or semantically correct.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: All claims made in the abstract and introduction are supported by results and analysis presented in the "Results" section in text and in the accompanying tables and figures. All results discussed are within the scope of the paper's main arguments.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We have included a separate "Limitations" section that outlines the limitations of this study, including a single clinical trial participant with chronic BCI implants, the limited availability of public datasets with high-SNR intracortical speech signals suitable for voice synthesis BCI, and the generalizability of our model to closed-loop BCI from our offline (simulated real-time) analysis.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: No theoretical results are presented or discussed in this paper.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: All methods and model architecture details are described in-depth in the "Methods" section, with citations to more detailed explanations where needed. Model architecture, training, and evaluation code is included in the supplemental material to reproduce results. We use publicly available benchmark intracortical neural dataset to evaluate our model and the citation to the dataset is provided. We are also providing voice samples synthesized by the model in the supplementary materials.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We have provided open access to all code including the model architecture, training, and evaluation code to reproduce results presented in the study. The data is also available publicly we have provided a reference to access this dataset.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Training details and data splits are described in the main text in the section "Model training". All model and training hyperparameters are included in the "Appendix".

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Error bars are reported for each analysis in the form of 95% confidence intervals derived by bootstrapping with replacement or with standard deviations where appropriate. No statistical comparisons were performed in this study.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Gerenal details on the computational requirements needed to train the model on this dataset and perform inference are included in the "Model training" section of the main text.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: All research conducted for this paper does conform to the NeurIPS Code of Ethics, as well as to the regulatory guidelines provided by the relevant institutional review board and the federal government.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: While we discuss societal impacts of speech BCI technology as a promising medical technology to restore lost motor function, we don't believe that the work presented in this paper on a benchmark dataset presents "broad" or "societal" impacts. This implanted brain-computer interface research is still in the early days of medical prostheses and very far from the potentially large-scale consumer use-cases that could involve negative societal impacts.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The models and data included with this paper pose no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: Existing code, data, and models that were used in this paper have been properly cited throughout the text (e.g., intracortical neural benchmark dataset, Whisper ASR model).

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: The code is well documented and other documentation is provided.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [Yes]

Justification: We performed intelligibility assessment of the voice neurally synthesized by our BCI model by asking human listeners to listen to the voice and transcribe it. We used a paid crowd sourcing platform to do human transcriptions. We have provided the instructions given to the transcribers in the "Appendix".

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [Yes]

    Justification: This study was approved by the appropriate institutional review board and federal government body and followed all corresponding guidelines appropriately. Additional details are provided in the methods section of the paper.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs were not used for any original or non-standard component of the core methods in this research. We have provided appropriate citation to the automatic speech recognition software we used as a secondary assessment of the BCI-synthesized voice.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.