Cut, Stitch, Adapt: Data-Efficient and Low-Calibration Speech BCIs

Animan Naskar*, Ankush Naskar*, and Deepti Bathula**

Department of Computer Science and Engineering,
Indian Institute of Technology Ropar, Rupnagar, Punjab, India
{2022csb1297, 2022csb1068}@iitrpr.ac.in, bathula@iitrpr.ac.in

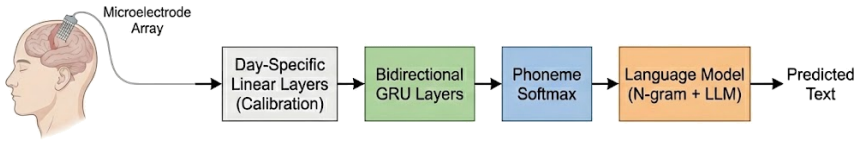
Abstract. Intracortical Brain–Computer Interfaces (iBCIs) for speech decoding show immense promise in translating neural activity into text. However, clinical viability remains severely constrained by labeled data scarcity and the continuous physical drift of neural representations over time. Contemporary approaches increasingly rely on massive cross-species pretraining and large ensembles, incurring prohibitive costs while still requiring daily recalibration to counteract electrode drift. We propose a data-efficient pipeline that requires only 40 hrs of patient data and achieves domain generalization to unseen recording sessions. To overcome data limitations, we propose **Neural Cut-and-Stitch (NCS)**, a signal-space augmentation strategy. By isolating word-level neural patterns from past recordings, we apply crossfade stitching to synthesize brain signals for entirely new sentences. We also introduce an **Adversarial Domain Adaptation (ADAN)** framework to improve generalization to unseen sessions. This explicitly enforces a session-invariant neural manifold, retaining acoustic decoding performance with far lesser calibration. Experiments on the Brain-to-Text '25 benchmark demonstrate that NCS alone reduces the Phoneme Error Rate (PER) from **10.1%** to **8.8%** on the held-out test set. Combining it with the baseline, we reduce the Word Error Rate (WER) from **7.0%** to **4.8%**. Concurrently, ADAN confines PER degradation on unseen sessions $\sim$ **15 days** into the future to $<$**2%** change. These gains show that signal augmentation and domain adaptation can be complementary to large-scale pretraining and reduce reliance on daily recalibration for deployment.

Keywords: Brain-Computer Interface · Domain Adaptation · Data Augmentation

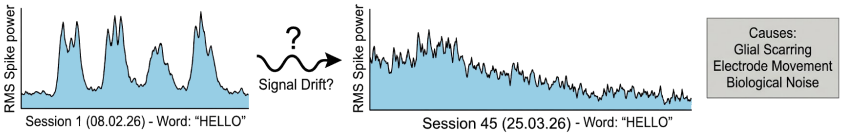
1 Introduction

Translating iBCIs from laboratory demonstrations to independent clinical use exposes two primary bottlenecks: the inherent non-stationarity of neural signals over time due to micro-motions and glial scarring, and the severe scarcity of training data relative to the combinatorial complexity of natural language. Recent

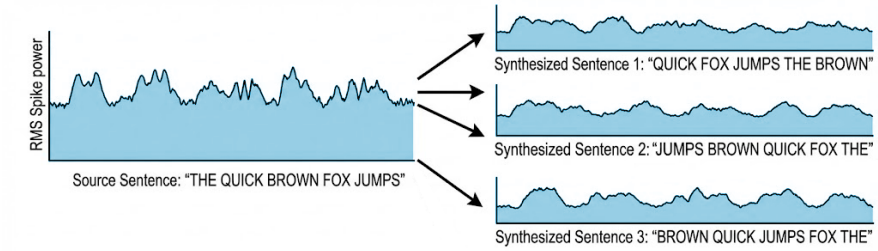
* Equal contribution.
** Corresponding author



(A) Baseline Pipeline: Translates continuous neural signals into text utilizing day-specific calibration layers and a recurrent acoustic decoder.



(B) Recording Drift: Unpredictable biological and mechanical factors alter neural distributions over time, breaking fixed calibration models.



(C) Neural Cut-and-Stitch: Synthesizing diverse, novel sentences by reordering existing recordings to directly combat severe vocabulary scarcity.

Fig. 1: **Fundamental Challenges in Speech iBCIs.** Overview of the baseline architecture alongside the primary bottlenecks addressed by our framework.

studies [5] demonstrate that pretraining on hundreds of hours of cross-task neural data, combined with massive autoregressive LLMs, yields exceptionally low Word Error Rates (WER). However, this magnitude of data and compute is inaccessible for standard clinical deployment. Furthermore, these large-scale models typically rely on within-day z-scoring. This normalization masks their inability to generalize to unseen future sessions. While data augmentation and domain adaptation have improved stability in cursor-control iBCIs [4] and non-invasive EEG [3], their application to high-dimensional, continuous speech decoding remains critically underexplored. We propose a practical alternative under strict data constraints by explicitly targeting data efficiency and representation stability. To tackle data scarcity, we introduce *Neural Cut-and-Stitch (NCS)*, a novel signal-space augmentation technique that synthesizes diverse training samples by re-ordering existing neural recordings. To resolve recording drift, we employ an *Adversarial Domain Adaptation (ADAN)* framework designed to align feature distributions across sessions, eliminating the dependency on daily calibration data. Prioritizing the low-latency footprint required for real-time prostheses, a

single Gated Recurrent Unit (GRU) trained with our NCS framework reduced the baseline Phoneme Error Rate (PER) from 10.1% to 8.8%. Concurrently, our ADAN framework successfully achieved domain generalization to uncalibrated future sessions, demonstrating distinct and practical pathways for reducing supervised recalibration during deployment.

Importantly, our contribution is not a fundamentally new neural decoding architecture, but a system-level strategy for improving data efficiency and reducing calibration in intracortical speech BCIs.

2 Methodology

2.1 Baseline Architecture

Our approach builds upon the two-stage decoding pipeline implementation released by the authors of [2].

A. Neural Decoding: Neural activity from the 256-channel array is binned into 20 ms non-overlapping windows. Extracting threshold crossings and spike-band power yields a high-dimensional feature vector $x_t \in \mathbb{R}^{512}$ at each timestep. These features are mapped to a probability distribution over 41 phoneme classes using a 5-layer bidirectional GRU (hidden size 1024), optimized via the Connectionist Temporal Classification (CTC) loss function. [11, 15]

B. Language Model Cascade: Text transcription is performed hierarchically. A 5-gram Viterbi beam search extracts the top-100 sentence hypotheses from the acoustic logits. These candidates are subsequently rescored using the pretrained `OPT-6.7B` Large Language Model. This model blends acoustic, N-gram, and semantic scores to select the final transcription. [13]

2.2 Neural Cut-and-Stitch Augmentation

The primary bottleneck in training robust neural decoders is the Zipfian distribution of labeled cortical data. An analysis of the Brain-to-Text '25 training set revealed severe sparsity: out of approximately 3,500 unique words, nearly 3,000 appear only once across the 9,000 training sentences. Consequently, the model overfits to specific neural contexts, failing to generalize when a known word appears in a novel phonetic neighborhood. To overcome this, we introduce *Neural Cut-and-Stitch (NCS)*:

A. Segmentation via Forced Alignment: To isolate word-level neural signatures, we overfit the baseline RNN on the training set to obtain frame-level phoneme alignments, observing a consistent structure:

$$\text{Signal} \approx (\text{BLANK}) \oplus [P_1, \ldots, P_n] \oplus (\text{SPACE}) \tag{1}$$

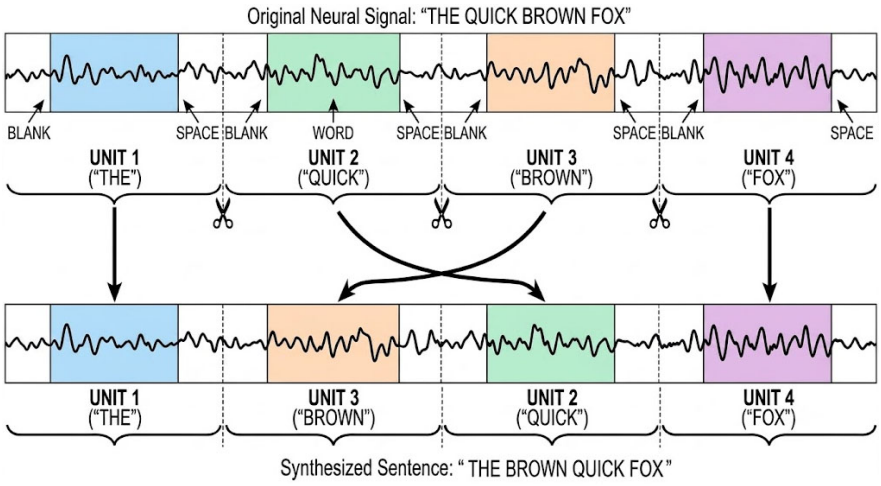


Fig. 2: **Neural Cut-and-Stitch Augmentation.** Continuous neural recordings are segmented into discrete word units, reordered to form novel sentences, and joined via spectral crossfading. This synthesizes contextually diverse training signals to effectively mitigate vocabulary scarcity.

Here, $\oplus$ denotes temporal concatenation and $P_1, \ldots, P_n$ the phoneme sequence of a word; `BLANK`/`SPACE` act as natural boundary delimiters. Using these boundaries, we built a dictionary mapping each word w to its observed neural tensors. Boundaries typically fall within long BLANK/SPACE regions, making minor cut-point errors label-preserving. All alignments and dictionary construction used training sessions only, with no statistics derived from validation or held-out future sessions.

B. Synthetic Corpus Generation: To artificially expand contextual diversity, we constructed a synthetic corpus comprising 10,000 new meaningful sentences (to model standard phonetic transitions) and 2,000 randomized word sequences (acting as a regularizer against semantic over-reliance). For each synthetic sentence, corresponding neural data was generated by iteratively sampling a candidate segment from $D(w)$ for each target word.

C. Signal Reconstruction: Naïve concatenation of neural segments introduces sharp boundary discontinuities that destabilize RNN hidden states. For discrete Threshold Crossings (TCs), we un-normalized the segments, concatenated the raw integers, and re-computed global normalization statistics to ensure a natural sequence distribution. For continuous Spike Band Power (SBP), we aligned the mean and variance statistics of adjacent segments prior to applying a brief temporal crossfade. This ensured smooth spectral continuity across transition boundaries.

2.3 Generative Synthesis of Neural Signals

While NCS synthesizes diverse training sentences by reordering word-level segments, we explored whether we could replicate natural co-articulatory transitions *across* word boundaries in synthetic sentences. To achieve this, we passed real recordings through the frozen baseline decoder, f_{dec}, extracting frame-level soft-logit sequences of corresponding words that capture phoneme durations and transitional uncertainties. We concatenated these to represent novel sentences and trained a bidirectional GRU generator, G, to invert them back into continuous neural signals, $\hat{x}$:

$$\hat{x} = G\left(\bigoplus_{i=1}^{N} f_{dec}(x_{w_i})\right) \tag{2}$$

However, this approach suffers from three compounding failure modes: the soft-logits carry the decoder's inherent 10.1% error; the CTC training signal depends on this same imperfect decoder; and Soft-DTW optimises a soft minimum over alignments rather than a single ground-truth path, introducing path-averaging spectral artifacts that the acoustic decoder subsequently overfit to. This branch was therefore omitted from the final pipeline in favor of the more stable NCS method. [12, 11]

2.4 Adversarial Domain Adaptation for Day-Agnostic Decoding

Longitudinal iBCI deployment is severely hindered by the continuous physical drift of the neural manifold. Standard mitigations employ day-specific linear input matrices; however, this fundamentally fails in real-world scenarios as it requires extensive recalibration data for every unseen future day. To achieve true domain generalization to unseen future sessions, we propose an Adversarial Domain Adaptation (ADAN) framework that enforces day-invariant acoustic features. Let $x \in \mathbb{R}^{T\times 512}$ be the neural input sequence, and $d \in \{1, \ldots, N\}$ the recording day. The architecture modifies the pipeline through a Min-Max game involving three components:

A. Day-Agnostic Encoder (f_{enc}): A shared feature extraction block comprising Group Normalization and linear projection, generating representations $h = f_{enc}(x; \theta_{enc})$. [14]

B. Statistical Day Classifier (f_{day}): To prevent destructive non-linear scrambling of the delicate phonetic geometry, we restrict the adversary's input to global sequence statistics (channel-wise temporal mean μ and standard deviation σ). The classifier predicts the recording day $\hat{d} = f_{day}(\mu(h), \sigma(h); \theta_{day})$.

C. Gradient Reversal Layer (GRL): Positioned just adjacent to the day classifier. During backpropagation, it negates standard cross-entropy gradients

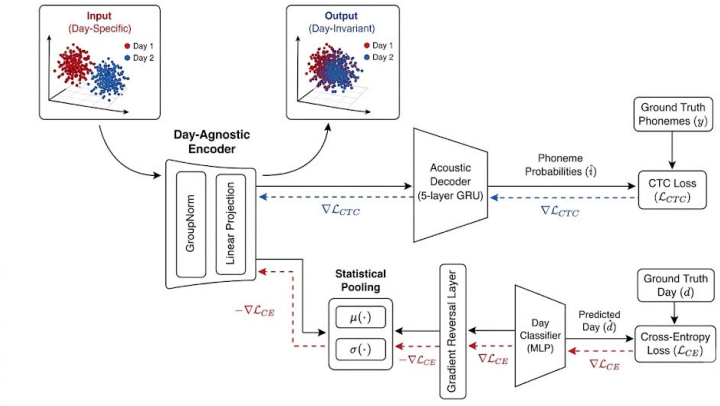


Fig. 3: **ADAN Architecture.** Neural features pass through a day-agnostic encoder whose output feeds both the acoustic decoder and a statistical day classifier. A Gradient Reversal Layer (GRL) penalizes preservation of day-specific artifacts, enforcing session-invariant features.

before passing them to the encoder, forcing f_{enc} to maximize the classifier's uncertainty. [10] The network is trained via the adversarial objective:

$$\min_{\theta_{\text{enc}},\theta_{\text{dec}}} \max_{\theta_{\text{day}}} \mathcal{L}_{\text{CTC}}(f_{\text{dec}}(h), y) - \mathcal{L}_{\text{CE}}(\hat{d}, d) \tag{3}$$

3 Data and Experimental Setup

3.1 The Brain-to-Text '25 Dataset

We evaluate on the *Brain-to-Text '25* benchmark, recorded from participant T15 with four high-density arrays (256 electrodes) in speech motor cortex. Each of the 45 official sessions provides train/validation/test splits; we merge train+validation within sessions for all experiments. We use: (1) **Standard:** official test sets; (2) **Chronological Future:** sessions ordered chronologically (∼15-day intervals), training on Sessions 1–44 (train+val) and holding out Session 45 test data as uncalibrated data. No augmentation, normalization statistics, or domain-adaptation components were fit using data from Session 45.

3.2 Implementation Details

Our implementation builds on the Python implementation released by the authors of [2] and is optimized for a low-resource setup (2× NVIDIA T4 GPUs). The optimal epoch count was determined on per-session validation splits in a preliminary run; models were then retrained on merged train + validation data, with Session 45 never used for tuning. Reported ± values denote standard deviation over three runs.

4 Results and Analysis

4.1 System-Level Comparison

Table 1 benchmarks our strategies against the baseline and massive-scale models. NCS reduces PER to 8.8%, while ADAN achieves 9.6% when trained on all sessions. In contrast, retraining the baseline solely on generated signals yields a degraded 21.9% PER. Since all methods use a frozen OPT-6.7B language model, performance differences arise entirely from the 30M-parameter phoneme decoder. Peak accuracy is obtained by ensembling the baseline with NCS (60M trainable parameters).

In contrast, models such as BIT [5] rely on cross-species pretraining and large ensembles, whereas ours are trained using participant-only data.

Table 1: **System-Level Comparison.** Our distinct approaches operate with a drastically smaller trainable footprint than concurrent scaling methods.

System	Architecture	Phoneme Decoder (trainable)	Total Training Data	PER	WER
Official Baseline	RNN + OPT-6.7B	30M	40 Hours	10.1%	7.0%
BIT [5]	Cascaded Ensemble	∼13M × 10	407 Hours	–	2.2%
Ours (Generative)	**Gen + OPT-6.7B**	**30M**	**0 Hours***	**21.9 ± 0.2%**	**10.1 ± 0.2%**
Ours (ADAN)	**ADAN + OPT-6.7B**	**30M**	**40 Hours**	**9.6 ± 0.2%**	**6.5 ± 0.2%**
Ours (NCS)	**NCS + OPT-6.7B**	**30M**	**40 Hours**	**8.8 ± 0.2%**	**5.5 ± 0.2%**
Ours (Ensemble)	**Base + NCS + OPT-6.7B**	**60M**	**40 Hours**	–	**4.8 ± 0.2%**

**Note: The phoneme decoder is trained entirely on synthetic signals produced by the generative model; the generative model itself was trained on the 40-hour real dataset.*

4.2 Generalization to Unseen Future Sessions

Session drift is known to degrade decoder performance over time [4]. We held out the final session entirely from training as a proxy for low-recalibration deployment (Table 2). The standard baseline suffers substantial degradation (+20.1% Δ PER) due to uncompensated temporal drift, while ADAN aligns the unseen manifold to its day-invariant feature space, degrading by only +1.5%.

Table 2: **Session-level domain generalization.** Degradation on Session 45 when held out of training.

Method	PER (Seen)	PER (Unseen)	Degradation (Δ)
Baseline (Day Matrix)	21.1 ± 0.2%	41.2 ± 0.2%	+20.1%
Only GRU	24.5 ± 0.2%	27.2 ± 0.2%	+2.7%
Ours (ADAN)	**19.1 ± 0.2%**	**20.6 ± 0.2%**	**+1.5%**

4.3 Longitudinal Stability and Day Drift

Longitudinal deployment demands consistent decoding accuracy despite neural signal non-stationarity over time. Figure 4 illustrates the daily validation PER when models are trained on the full dataset (all sessions seen). While the standard Day Matrix baseline and shared-input GRU exhibit noticeable performance degradation during late-stage signal drift, our proposed NCS augmentation establishes a consistently lower error floor across the entire recording timeline.

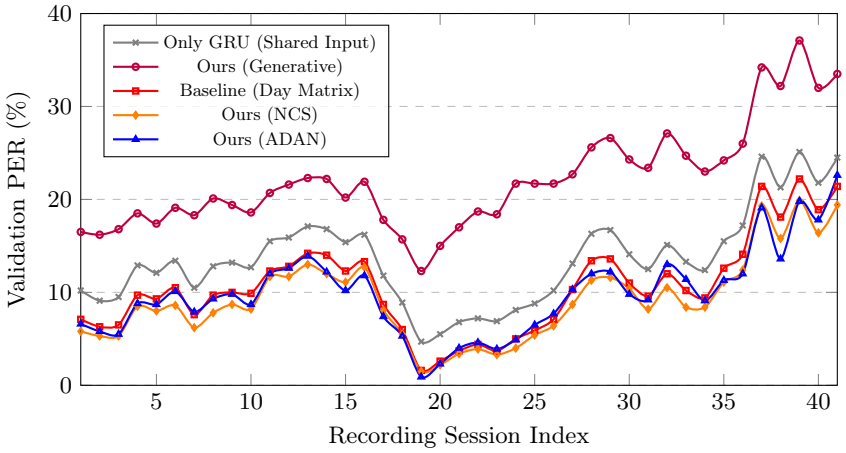


Fig. 4: **Day Drift (all sessions are seen).** Validation PER across the recording timeline (smoothed using 3-session moving average for visual clarity). The NCS model consistently operates at a lower error floor across the recording period.

5 Conclusion

We address neural data scarcity and longitudinal signal drift in clinical speech iBCIs. Our Neural Cut-and-Stitch (NCS) augmentation improves PER and WER under strict participant-specific data constraints, while our Adversarial Domain Adaptation (ADAN) framework mitigates degradation on chronologically unseen sessions by enforcing a day-invariant feature space.

Together, these complementary strategies provide a practical pathway toward stable, real-time speech neuroprostheses under clinical deployment constraints, where data collection is limited and frequent recalibration is burdensome.

Code Availability

Code is available at https://github.com/Miccai-26/brain-to-text-ncs-adan.

Disclosure of Interests

The authors have no competing interests to declare.

References

1. Willett, F.R., Kunz, E.M., Fan, C., et al.: A high-performance speech neuroprosthesis. Nature **620**(7976), 1031–1036 (2023).
2. Card, N.S., Wairagkar, M., Iacobacci, C., Hou, X., Singer-Clark, T., Willett, F.R., Kunz, E.M., Brandman, D.M., et al.: An accurate and rapidly calibrating speech neuroprosthesis. *New England Journal of Medicine* (2024). Code available at: https://github.com/Neuroprosthetics-Lab/nejm-brain-to-text.
3. Zanini, P., Congedo, P., Jutten, C., Said, S., Berthoumieu, Y.: Transfer learning: A Riemannian geometry framework with applications to brain-computer interfaces. IEEE Transactions on Biomedical Engineering **65**(5), 1107–1116 (2018).
4. Degenhart, A.D., Bishop, W.E., Oby, E.R., et al.: Stabilization of a brain-computer interface via the alignment of low-dimensional spaces of neural activity. Nature Biomedical Engineering **4**(7), 672–685 (2020).
5. Zhang, Y., He, L., et al.: A cross-species neural foundation model for end-to-end speech decoding. arXiv preprint (2025).
6. Moses, D.A., Leonard, M.K., et al.: Neuroprosthesis for Decoding Speech in a Paralyzed Person. *New England Journal of Medicine* **385** (2021).
7. Anumanchipalli, G.K., Chartier, J., Chang, E.F.: Speech synthesis from neural decoding of spoken sentences. *Nature* **568**, 493–498 (2019).
8. Pasley, B.N., David, S.V., Mesgarani, N., et al.: Reconstructing speech from human auditory cortex. *PLoS Biology* **10**, e1001251 (2012).
9. Herff, C., Schultz, T.: Generating natural, intelligible speech from brain activity: A review of decoding approaches in ECoG and intracranial recordings. *Frontiers in Neuroscience* (2019).
10. Ganin, Y., Lempitsky, V.: Domain-adversarial training of neural networks. *Journal of Machine Learning Research (JMLR)* **17** (2016).
11. Graves, A., Fernández, S., Gomez, F., Schmidhuber, J.: Connectionist Temporal Classification: labelling unsegmented sequence data with recurrent neural networks. In: Proceedings of ICML (2006).
12. Cuturi, M., Blondel, M.: Soft-DTW: a differentiable loss function for time-series. In: Proceedings of ICML (2017).
13. Zhang, S., Roller, S., Goyal, N., et al.: OPT: Open Pre-trained Transformer Language Models. arXiv preprint (2022).
14. Wu, Y., He, K.: Group Normalization. In: Proceedings of the European Conference on Computer Vision (ECCV) (2018).
15. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Computation* **9**(8), 1735–1780 (1997).