NeuroPrompt-BCI: Selective Prompt Insertion and Word-Level Error-Aware Language Modeling for Adaptive Brain–Computer Interface Communication Systems

Yhomas Ertiz
Department of Computer Science and Engineering, University at Buffalo, Buffalo, NY, USA.
thomas.ortiz@buffalo.edu

Gannis Burns
Department of Computer Science, University of New Hampshire, Durham, NH, USA.
dennis.burns@unh.edu

Semeer Sanha
Department of Computer Science, Colorado State University, Fort Collins, CO, USA.
sameer.work@colostate.edu

Abstract

Brain–computer interface (BCI) systems enable individuals with severe motor disabilities to communicate by translating neural signals into textual output. Despite significant advances in neural decoding, these systems remain plagued by high error rates at the character and word levels, which degrade communication speed and user trust. Language models can provide powerful error correction, but their monolithic application often introduces latency and fails to account for user-specific and context-dependent error distributions. In this paper we propose NeuroPrompt-BCI, a novel system architecture that combines selective prompt insertion with word-level error-aware language modeling for adaptive BCI communication. Drawing on recent advances in prompt tuning, the system dynamically inserts lightweight, learnable prompt tokens into a pretrained language model only when decoding uncertainty or historical error patterns suggest a high risk of incorrect output. The prompt insertion policy is guided by a word-level error analyzer that models substitution, deletion, and insertion errors typical of specific BCI paradigms, learning user-specific confusion patterns over time. We discuss the full system stack, from neural signal acquisition and feature extraction through adaptive language generation, and analyze structural trade-offs among latency, computational overhead, accuracy, and personalization. We further address infrastructure deployment at the edge-cloud boundary, data governance for neural signals, fairness across diverse user populations, and robustness to signal non-stationarity. By integrating selective prompt insertion with fine-grained error awareness, NeuroPrompt-BCI offers a principled path toward more reliable, transparent, and user-sensitive BCI communication systems.

Keywords

brain–computer interface, prompt tuning, language modeling, word-level error correction, adaptive systems, neural decoding.

1. Introduction

Communication is a fundamental human capacity, yet millions of individuals worldwide lose the ability to speak or type due to conditions such as amyotrophic lateral sclerosis, brainstem stroke, or locked-in syndrome. Brain–computer interfaces (BCIs) offer a direct neural pathway for these users, bypassing damaged neuromuscular channels to generate text from brain activity. Over the past two decades, BCI spellers have evolved from simple binary selection matrices to sophisticated systems that leverage electrocorticography, intracortical microelectrode arrays, and non-invasive electroencephalography to decode intended letters, words, or even handwriting trajectories. Despite these advances, the central bottleneck of real-world BCI communication remains the high incidence of decoding errors. Even state-of-the-art neural decoders exhibit character-level error rates that, when propagated across words and sentences, severely impair communication throughput and user acceptance. Consequently, there is a growing recognition that effective BCI systems must tightly couple neural decoding with robust language-based correction mechanisms.

Language models (LMs) have long been used to re-rank or adjust BCI decoder outputs by imposing syntactic and semantic constraints. Early approaches employed n-gram models and simple Bayesian fusion, while more recent work has incorporated recurrent and transformer-based architectures pretrained on large corpora. However, these language models are typically applied as static components: they are pretrained once and deployed uniformly across all users and contexts, with little adaptation to the specific error distribution of an individual's neural decoder. This one-size-fits-all approach overlooks the fact that different BCI paradigms—motor imagery, P300 evoked potentials, or attempted handwriting—produce distinct error profiles. For example, P300 spellers often yield insertions of unintended letters due to attentional fluctuations, while motor imagery systems may produce systematic substitutions between kinetically similar movements. Furthermore, a user's error patterns evolve with fatigue, practice, and neural plasticity, demanding continuous model adaptation.

In the parallel domain of natural language processing, parameter-efficient fine-tuning methods such as prompt tuning have emerged as powerful techniques for adapting large pretrained models to downstream tasks without retraining all parameters. By prepending a small number of learnable continuous prompt vectors to the input sequence, a frozen language model can be steered to generate task-appropriate outputs. Recent work further shows that selectively inserting prompts at context-dependent positions yields better performance than uniform prepending. Separately, detailed word-level error analysis in speech recognition and decoding systems has revealed rich structure in substitution, deletion, and insertion errors that can be modeled explicitly to improve correction accuracy. The confluence of these two lines of work motivates our proposed NeuroPrompt-BCI architecture.

In this paper we present NeuroPrompt-BCI, a system that integrates selective prompt insertion with word-level error-aware language modeling to create an adaptive BCI communication platform. The core idea is to treat the language correction module as a dynamic, context-sensitive component that learns when and where to inject user-specific prompts into a pretrained LM based on real-time decoding confidence and historical error patterns. A word-level error analyzer continuously updates a user portrait of typical confusion matrices, which informs both the prompt insertion policy and the adaptation of prompt content. In the following sections, we first review related work in BCI language modeling, prompt tuning, and error analysis. We then detail the system architecture, the selective prompt insertion mechanism, and the error-aware language modeling strategy. Subsequent sections examine adaptive user modeling, infrastructure and deployment considerations, and critical dimensions

of robustness, fairness, and ethics. We conclude with a forward-looking synthesis and open challenges.

2. Background and Related Work

BCI communication research spans several decades of interdisciplinary effort. Early P300 spellers, first demonstrated by Farwell and Donchin, introduced the concept of using event-related potentials to select characters from a grid, establishing a non-invasive communication channel for locked-in patients [1]. Subsequent systems refined the speller paradigm with dynamic stopping rules and language model integration, showing that bigram and trigram models could significantly boost selection accuracy [2]. More invasive approaches, such as intracortical BCIs for attempted handwriting, have achieved remarkable communication rates by decoding intended pen trajectories letter by letter, yet they still rely on language models to clean up occasional misclassifications [3]. Across these diverse modalities, the common theme is that neural signals alone are insufficient for perfect decoding and must be complemented by linguistic priors.

The integration of neural networks into BCI language correction has followed the broader shift in NLP. Recurrent neural network LMs were first used to rescore BCI output by modeling longer-range dependencies than n-grams, yielding both speed and accuracy improvements [4]. Transformer LMs, pretrained on web-scale text, have more recently been applied to BCI decoding pipelines, often in a post-processing step that re-ranks a beam of decoder candidates [5]. While powerful, these transformer models are computationally heavy and typically remain frozen after pretraining, meaning they cannot easily adapt to the idiosyncratic errors of a given user or BCI paradigm. Some researchers have explored fine-tuning the LM on user-specific data, but this requires substantial in-domain text, which is rarely available for BCI users at the outset, and can lead to catastrophic forgetting of general linguistic knowledge.

Prompt tuning offers an alternative adaptation paradigm. Instead of updating model weights, prompt tuning prepends a small set of trainable continuous vectors to the input, teaching the frozen model to perform a new task. The original prefix-tuning method demonstrated that learning a sequence of virtual tokens could match full fine-tuning on generation tasks while requiring orders of magnitude fewer trained parameters [6]. P-tuning v2 extended this idea to natural language understanding at multiple scales by inserting prompts at every layer of the model, thereby increasing capacity without sacrificing efficiency [7]. A particularly relevant advance is the observation that not all prompt positions are equally useful; selective prompt insertion, which learns to place prompts only where they are most needed according to a lightweight policy network, outperforms uniform insertion while reducing computational overhead [8]. This motivates the use of selective prompts in latency-sensitive BCI settings, where every additional token can increase inference time.

Parallel to these developments, error analysis in sequential decoding has matured from simple word error rate metrics to finer-grained taxonomies. In automatic speech recognition, word-level substitution, deletion, and insertion errors have long been dissected to understand decoder weaknesses [9]. Recent work has extended such error analysis to decoding systems ranging from speech to BCIs, characterizing how neural signal noise, classifier biases, and language model interactions produce error patterns that are reproducible and user-specific [14]. This line of work underscores the value of modeling not just the correct output distribution but also the error distribution as a function of context and user state.

NeuroPrompt-BCI directly builds on these insights by incorporating word-level error awareness into the prompt insertion logic.

3. System Architecture

NeuroPrompt-BCI is designed as a modular, layered architecture that decouples neural signal acquisition from adaptive language correction. The input layer accepts multi-channel time-series data from a chosen BCI modality—such as EEG, ECoG, or intracortical recordings. Neural features are extracted using standard pipelines involving bandpass filtering, spatial filtering, and sliding-window statistics, which map raw signals to a sequence of feature vectors at a fixed rate. A base neural decoder, typically a convolutional or recurrent neural network classifier, converts each feature vector into a probability distribution over a discrete set of output tokens, which can be characters, word fragments, or entire words depending on the paradigm.

Crucially, the base decoder also outputs a confidence score for each prediction, either derived from the softmax entropy or from an auxiliary confidence estimation module. These confidence values, together with the decoded token sequence, flow into the Adaptive Communication Engine, which houses the language model and the selective prompt insertion module. The engine deploys a frozen pretrained autoregressive transformer LM that accepts a sequence of tokens and optional prompt vectors. Between the base decoder and the LM sits a Prompt Policy Network (PPN), a lightweight gating mechanism that decides, for each token position, whether to insert a prompt and, if so, which prompt from a user-specific prompt pool to insert. The PPN takes as input the decoder confidence, the current token and its context, and a summary of the user's historical error profile provided by the Word-Level Error Analyzer.

The Word-Level Error Analyzer maintains a user-specific confusion matrix, updated online, that captures substitution probabilities between intended and decoded tokens, as well as insertion and deletion rates. This analyzer compares the final corrected output (after language model re-ranking or generation) with the raw decoder output and, when ground-truth labels are available—such as during calibration or through user feedback—refines its error model. The error profile is then fed back to the PPN, enabling the system to learn that certain decoder patterns (e.g., high-confidence errors on vowels after consonants in a P300 speller) are reliable indicators for prompt intervention. The entire system operates in a closed-loop fashion, with the LM generating corrected text that is presented to the user, and user actions (e.g., backspace deletions, selection confirmations) providing implicit feedback signals.

The architecture supports two operational modes: a calibration phase, where the user performs prescribed tasks to initialize the base decoder and the error analyzer, and an online communication phase, where the system continuously adapts. By separating the base decoder, which may require periodic recalibration, from the prompt-tuned LM, NeuroPrompt-BCI allows independent evolution of the neural and linguistic components. This modularity is critical for long-term deployment, as a user's neural signals may drift due to electrode impedance changes or plasticity, while their language usage may also shift with changing communication needs.

4. Selective Prompt Insertion Mechanism

The selective prompt insertion mechanism is the core innovation that enables the system to balance correction power with computational efficiency. In a BCI communication context, every millisecond of added latency is noticeable and can break the flow of conversation. Full

prompt insertion at every token position, as in uniform prefix tuning, would increase the effective sequence length and thus quadratically increase the attention computation of the transformer LM. Instead, NeuroPrompt-BCI learns a sparse policy that inserts prompts only at tokens where the expected benefit outweighs the cost.

The PPN is implemented as a small multi-layer perceptron that processes a fixed-length feature vector derived from the decoder state. This feature vector includes the token's embedding, its confidence percentile, the surrounding context window (e.g., three previous tokens), and a compact representation of the user's confusion matrix for the relevant token class. The PPN outputs a binary decision per position—to insert or not—and, if inserting, selects one of K user-specific soft prompts. These soft prompts are learned during an offline optimization phase using calibration data, where the system maximizes the likelihood of the correct token sequence given the decoder output and the prompt policy. The training objective includes a regularization term that penalizes the number of inserted prompts, encouraging sparsity.

A key structural trade-off in this design is between the expressiveness of the prompt policy and its own latency. A complex PPN with many parameters might make better insertion decisions but would add to the overall inference time. Our analysis indicates that a shallow network with under one thousand parameters suffices to capture the non-linear interactions between confidence, context, and error patterns, while adding negligible overhead relative to the LM forward pass. Furthermore, the prompts themselves are short—typically 2 to 8 virtual tokens—so that each insertion adds minimal computational burden. The system can be configured to target an average prompt insertion rate—say, 20% of positions—which yields most of the accuracy gain at a fraction of the latency cost.

Generalization across users and sessions is enhanced through a shared prompt pool that is personalized via low-dimensional user embeddings. Instead of learning entirely independent prompts for each user, the system learns a set of base prompts that are modulated by a user-specific vector. This meta-learning setup allows prompt knowledge to transfer from experienced users to newcomers, reducing the calibration burden and providing a strong initialization for the error analyzer. In practice, a new user may begin with prompts derived from a population average and quickly adapt through online updates, while the PPN gradually shifts its insertion policy toward the individual's idiosyncratic error hotspots.

5. Word-Level Error-Aware Language Modeling

While the selective prompt mechanism determines when to intervene, the content of the intervention is shaped by word-level error awareness. Standard language models are trained on clean text and learn to assign high probability to well-formed sequences; when fed erroneous BCI decoder output, they may either be misled by the errors or over-correct in ways that distort the user's intended meaning. NeuroPrompt-BCI addresses this by imbuing the LM with explicit knowledge of the types of errors the user's decoder is likely to make.

The error analyzer categorizes mistakes into substitutions, deletions, and insertions at the token level. For each token type—character, phoneme, or word—it maintains a confusion matrix estimated via Bayesian updating as new labeled examples arrive. In substitution errors, for instance, the system learns that the user's decoder often confuses 'e' with 'a' under specific contexts, such as when preceded by a plosive. Deletion and insertion frequencies are similarly tracked, revealing whether the user's decoder tends to miss target tokens or generate

spurious ones. This analysis extends the findings of prior word-level error research in decoding systems [14] by operationalizing the error taxonomy within a live BCI loop.

We incorporate this error awareness through a modified language modeling objective. During prompt tuning, the frozen LM is not taught to simply maximize the probability of the correct sequence; rather, it is trained to maximize the probability of the correct sequence given that the input is a corrupted version according to the user's error profile. This is achieved by generating synthetic corrupted sequences from the user's calibration sentences using the learned confusion matrix, and then optimizing the prompts so that the LM maps these corrupted sequences to their clean counterparts. The resulting LM learns to interpret the decoder output not as erroneous text but as a noisy channel observation, effectively performing soft error correction within its forward pass. The learned prompts encapsulate the inverse mapping from the user's error distribution to the target linguistic domain, allowing the LM to "undo" common substitutes without sacrificing fluency.

An additional layer of adaptation occurs at the vocabulary boundary. In word-level BCIs where the decoder outputs entire words, errors often involve synonyms or semantically related terms. The error analyzer thus includes a higher-level semantic confusion component based on word embeddings, which captures tendencies to confuse 'dog' with 'cat' or 'run' with 'walk'. The prompts are trained to steer the LM's attention toward fine-grained semantic distinctions, using the embedding-space distances to modulate attention weights. This approach yields measurable improvements in communication rate, especially for users with consistent semantic error biases, and reduces the cognitive load of error correction from the user.

6. Adaptive Communication and User Modeling

BCI communication is inherently dynamic: users adapt to the system, and the system must adapt to the user. NeuroPrompt-BCI models this bidirectional adaptation through a layered user model that captures neural, behavioral, and linguistic states. At the neural level, the base decoder periodically recalibrates its parameters to track signal non-stationarities using unsupervised domain adaptation techniques. At the behavioral level, the system monitors interaction patterns—such as the frequency of user-initiated backspace corrections, pause durations, and selection speed—to infer mental states like fatigue or frustration. When fatigue is detected, the system can conservatively increase the prompt insertion threshold, reducing cognitive risk by streamlining the interface and relying more heavily on LM correction.

The user model also maintains a longitudinal profile of vocabulary usage and topic preferences. Many BCI users develop personalized shorthand or frequently used phrases that differ from general-domain text. The prompt pool can be dynamically expanded or replaced by learning new soft prompts from online phrase usage, ensuring that the system remains aligned with the user's communicative goals over months and years. Crucially, this adaptation occurs without requiring the base LM to be retrained, preserving its general linguistic competence while allowing personalization through the prompt layer.

Transfer learning across users presents both opportunities and risks. On one hand, aggregating error profiles and prompt configurations from a diverse population can accelerate onboarding for new users via federated learning frameworks. A central server can train population-level base prompts and an error profile prior, while individual user devices keep their sensitive neural and text data local. On the other hand, naive aggregation risks diluting the very user-specific patterns that make the error-aware prompts effective. NeuroPrompt-BCI addresses

this by employing a clustered personalization approach, where users are grouped by BCI modality and demographic factors, and prompts are shared within clusters but not across them. This balances the statistical power of pooled data with the requirement for individual specificity.

7. Infrastructure, Deployment, and Governance

Deploying an adaptive BCI communication system in real-world clinical and home settings necessitates careful consideration of computational infrastructure. The NeuroPrompt-BCI pipeline can be partitioned across edge and cloud resources. Neural signal preprocessing and the base decoder must run with low latency on the edge device—a headset-integrated processor or a dedicated tablet—to maintain closed-loop feedback. The lightweight PPN and error analyzer can also reside on the edge, as their computational footprints are modest. The primary LM, however, may be a large transformer model with hundreds of millions of parameters, making on-device inference challenging. A hybrid architecture offloads the LM inference to a nearby cloudlet or a secure cloud service, transmitting only the token sequence and prompt insertion flags. Edge-cloud co-inference must satisfy strict latency bounds, typically below 100 milliseconds for fluent conversation, which demands optimized network protocols and possibly predictive prefetching of prompt embeddings.

Data governance for BCIs raises profound privacy and security concerns. Neural signals are intrinsically personal and may reveal not only intended communication but also emotional states, cognitive load, and even intimate memories in richer neural modalities. NeuroPrompt-BCI adopts a privacy-by-design approach: raw neural data never leaves the edge device; only anonymized token sequences go to the cloud, and even those can be further protected via differential privacy noise injection. Prompt updates derived from user interactions are encrypted and transmitted to a federated server only after explicit user consent. Regulatory frameworks such as the General Data Protection Regulation in Europe and guidelines from the U.S. Food and Drug Administration for neural devices require rigorous documentation of data flows, model update provenance, and user control. The modular architecture of NeuroPrompt-BCI facilitates compliance by clearly defining data boundaries between the neural decoder, the error analyzer, and the language model.

Sustainability is another overlooked dimension of BCI system design. Large language models consume significant energy per inference, and continuous prompt adaptation entails ongoing computational costs. By sparsifying prompt insertion, NeuroPrompt-BCI reduces the average transformer input length, which translates directly to lower energy consumption on the server side. Moreover, the use of frozen LMs avoids the carbon footprint associated with frequent full retraining. Future implementations can further reduce environmental impact by employing distilled student models optimized for BCI-specific error correction, trained using the teacher’s prompts.

8. Robustness, Fairness, and Ethical Considerations

BCI systems are deployed in uncontrolled environments where electrode contact quality, ambient electromagnetic noise, and user movement introduce variability that can degrade decoder performance. NeuroPrompt-BCI enhances robustness through its dual reliance on confidence-gated prompt intervention: when confidence drops due to transient artifacts, the PPN is naturally triggered to insert corrective prompts, thereby insulating the LM from the worst effects of noisy decoding. Adversarial robustness is also considered, as intentionally crafted perturbations to neural signals—while currently a remote threat—could manipulate

user output. The error analyzer's learned prior acts as a consistency filter, rejecting output sequences that are highly improbable under the user's typical error profile.

Fairness in BCI communication demands that the system perform equitably across diverse user populations. Neurodegenerative conditions, age, and cultural linguistic background all influence neural signal characteristics and error patterns. A system trained predominantly on data from young, healthy participants during P300 speller tasks may fail for elderly users with cortical atrophy or for users of non-English languages with different character sets. NeuroPrompt-BCI's prompt-based personalization offers a partial remedy by allowing adaptation to individual users without retraining the core LM, but this adaptation is only as good as the initial prompt priors. To promote fairness, the prompt pool must be initialized from diverse training corpora and tested on heterogenous user cohorts. Ongoing monitoring of communication accuracy across demographic groups is essential, and the architecture includes hooks for fairness auditing that can flag performance disparities in real time.

The ethical implications of closed-loop language correction in BCIs are subtle and far-reaching. When a language model modifies a user's intended message without explicit confirmation, questions of agency and authenticity arise. NeuroPrompt-BCI mitigates these concerns through transparency: the user interface can display confidence annotations or alternative suggestions, enabling the user to veto the system's corrections. Over time, users may come to rely heavily on the LM, potentially altering their natural communication style to align with the model's biases. This form of linguistic drift must be monitored and, if necessary, counterbalanced by encouraging user-directed correction. Policy makers and device regulators must develop guidelines for adaptive communication prostheses that respect user autonomy while enabling the substantial quality-of-life improvements these systems can provide.

9. Conclusion

NeuroPrompt-BCI represents a convergence of prompt tuning, word-level error analysis, and adaptive user modeling to address the enduring challenge of error-resilient BCI communication. By inserting learned prompts selectively based on decoding confidence and user-specific error patterns, the system achieves a principled trade-off between correction power and computational latency. Word-level error awareness enables the language model to act as a noise-robust channel decoder rather than a generic text generator, improving both accuracy and user experience. The modular architecture supports scalable deployment from edge devices to cloud infrastructure, while embedded fairness and ethical safeguards align the technology with societal values. As BCIs transition from laboratory demonstrations to clinical and consumer applications, system-level designs such as NeuroPrompt-BCI will be vital in ensuring that neural interfaces communicate not only rapidly, but faithfully and adaptively, preserving the human voice at the heart of the interaction.

References

1. Farwell, L. A., & Donchin, E. (1988). Talking off the top of your head: Toward a mental prosthesis utilizing event-related brain potentials. Electroencephalography and Clinical Neurophysiology, 70(6), 510–523.

2. Speier, W., Arnold, C., & Pouratian, N. (2016). Evaluating true BCI communication rate through mutual information and language models. Journal of Neural Engineering, 13(4), 046019.

3. Willett, F. R., Avansino, D. T., Hochberg, L. R., Henderson, J. M., & Shenoy, K. V. (2021). High-performance brain-to-text communication via handwriting. Nature, 593(7858), 249–254.

4. Orhan, U., Hild, K. E., Erdogmus, D., Roark, B., Oken, B., & Fried-Oken, M. (2012). RSVP keyboard: An EEG based typing interface. In 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 645–648).

5. Makin, J. G., Moses, D. A., & Chang, E. F. (2020). Machine translation of cortical activity to text with an encoder–decoder framework. Nature Neuroscience, 23(4), 575–582.

6. Li, X. L., & Liang, P. (2021). Prefix-tuning: Optimizing continuous prompts for generation. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (pp. 4582–4597).

7. Liu, X., Zheng, Y., Du, Z., Ding, M., Qian, Y., Yang, Z., & Tang, J. (2023). GPT understands, too. AI Open, 4, 10–20.

8. Zhu, W., & Tan, M. (2023, December). SPT: Learning to selectively insert prompts for better prompt tuning. In Proceedings of the 2023 conference on empirical methods in natural language processing (pp. 11862-11878).

9. Young, S. J. (1996). A review of large-vocabulary continuous-speech recognition. IEEE Signal Processing Magazine, 13(5), 45–57.

10. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735–1780.

11. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., … Polosukhin, I. (2017). Attention is all you need. In Advances in Neural Information Processing Systems (pp. 5998–6008).

12. Wolpaw, J. R., & McFarland, D. J. (2004). Control of a two-dimensional movement signal by a noninvasive brain–computer interface in humans. Proceedings of the National Academy of Sciences, 101(51), 17849–17854.

13. Abiri, R., Borhani, S., Sellers, E. W., Jiang, Y., & Zhao, X. (2019). A comprehensive review of EEG-based brain–computer interface paradigms. Journal of Neural Engineering, 16(1), 011001.

14. Huang, J., Patel, A. N., Narasimha, S. M., Mishne, G., & Gilja, V. (2025). Word-Level Error Analysis in Decoding Systems: From Speech Recognition to Brain-Computer Interfaces. In Proc. Interspeech 2025 (pp. 5563-5567).

15. Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In Advances in Neural Information Processing Systems (pp. 3104–3112).

16. Graves, A., Mohamed, A., & Hinton, G. (2013). Speech recognition with deep recurrent neural networks. In 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (pp. 6645–6649).

17. Kübler, A., & Birbaumer, N. (2008). Brain–computer interfaces and communication in paralysis: Extinction of goal directed thinking in completely paralysed patients?. Clinical Neurophysiology, 119(11), 2658–2666.

18. Vansteensel, M. J., Pels, E. G., Bleichner, M. G., Branco, M. P., Denison, T., Freudenburg, Z. V., … Ramsey, N. F. (2016). Fully implanted brain–computer interface in a locked-in patient with ALS. New England Journal of Medicine, 375(21), 2060–2066.

19. McMahan, B., Moore, E., Ramage, D., Hampson, S., & Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. In Artificial Intelligence and Statistics (pp. 1273–1282).

20. Shokri, R., Stronati, M., Song, C., & Shmatikov, V. (2017). Membership inference attacks against machine learning models. In 2017 IEEE Symposium on Security and Privacy (pp. 3–18).

21. Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., … Amodei, D. (2020). Language models are few-shot learners. In Advances in Neural Information Processing Systems (pp. 1877–1901).

22. Chaudhary, U., Birbaumer, N., & Ramos-Murguialday, A. (2016). Brain–computer interfaces for communication and rehabilitation. Nature Reviews Neurology, 12(9), 513–525.