DANCE: DETECT AND CLASSIFY EVENTS IN EEG

Anonymous authors
Paper under double-blind review

ABSTRACT

Event identification in continuous neural recordings is a critical task in neuroscience. Decoding in EEG is dominated by classifying windows aligned to known event onsets. However, while available in controlled experiments, such onsets are absent in continuous real-world monitoring. Here, we introduce DANCE, a deep learning pipeline that frames neural decoding as a set-prediction problem and jointly detects and classifies events directly from raw, unaligned signals. Evaluated separately on ten datasets curated from the literature with a wide variety of event types (ranging from milliseconds to minutes in duration), our model outperforms existing methods on a broad range of cognitive, clinical and BCI tasks. This single architecture establishes a new state of the art in the competitive task of seizure monitoring and matches the accuracy of onset-informed models for BCI tasks. Overall, our method marks a step towards end-to-end asynchronous neural decoding models.

1 INTRODUCTION

Temporal events are the primary unit of analysis in experimental and clinical neuroscience, from motor-imagery trials (Faller et al., 2012; Tangermann et al., 2012) and visual stimuli (Congedo et al., 2011; Korczowski et al., 2019a;b) to speech utterances (Accou et al., 2024; Broderick et al., 2018) and epileptic seizures (Shah et al., 2018). Beyond controlled environments, applications such as real-time neuroprosthetics (Willett et al., 2021), continuous wheelchair control (Zhang et al., 2024) and naturalistic neurofeedback (Le Franc et al., 2022) require moving from discrete event analysis to asynchronous, continuous decoding.

Despite this need, EEG decoding remains constrained by its inherently low signal-to-noise ratio (SNR) (Mak & Wolpaw, 2009). To mitigate it, the dominant practice is to evaluate models on pre-segmented windows time-locked to known event onsets, bypassing detection to focus purely on classification (Chevallier et al., 2024; Défossez et al., 2023; Banville et al., 2025; Tang et al., 2022; Lévy et al., 2025). While useful for controlled comparisons, this onset-informed paradigm does not reflect realistic settings where precise markers are unavailable. Existing asynchronous alternatives inherit well-documented failure modes (Dan et al., 2025; Chavelli et al., 2025): applying a classifier to every window produces fragmented predictions that require post-processing (Craik et al., 2019), while two-stage cascades that first detect then classify collapse when the no-event class dominates (Wu et al., 2024). A few models do predict events directly, for micro-sleep (Chambon et al., 2019), eye-tracking (Wolf et al., 2022), or artefacts and seizures (Seeuws et al., 2024), but each remains confined to a single domain. A unified framework for asynchronous EEG decoding is therefore still missing.

Event detection can be framed in two ways (Figure 1B): as a *dense* (sample-based) task, predicting event presence at every time sample, or as a *sparse* (event-based) task, predicting directly the set of events in a window. In computer vision, moving from the dense to the sparse view drove major progress in object detection (Girshick et al., 2014; Ren et al., 2015; Redmon et al., 2016), most notably by reframing detection as set prediction with the DETR family (Carion et al., 2020), evaluated with object-level metrics such as Intersection over Union (IoU; Maier-Hein et al. 2024). Temporal action detection extends this by combining set-based objectives with dense auxiliary supervision for more stable training (Zong et al., 2023; Wang et al., 2025b). Transferring these methods directly to scalp EEG nonetheless fails, because of low SNR, limited per-dataset sample size, and event durations spanning four orders of magnitude.

We introduce **DANCE**, an end-to-end framework that detects and classifies discrete events from raw EEG without pre-aligned windows. A Perceiver-based latent resampling module (Jaegle et al., 2021)

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

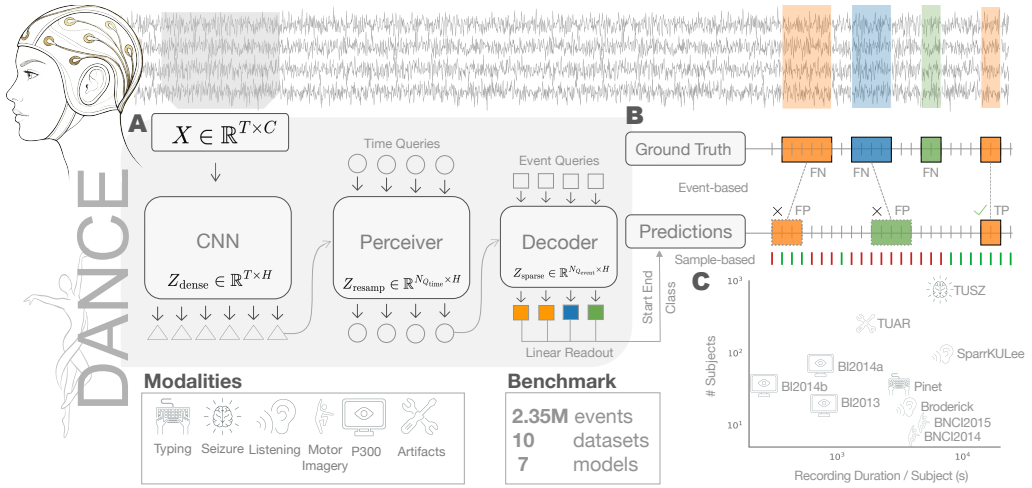

Figure 1: **Our approach. A.** The DANCE model processes EEG time-series windows of fixed length through a CNN backbone, a Perceiver and a Decoder. **B.** DANCE predicts event instances of different classes (here, represented by different colors), which are evaluated against ground truth using both event-based and sample-based metrics. Dashed contours represent incorrect events according to the event-based metric. **C.** The model is trained separately on 10 heterogeneous datasets comprising a total of 2.35 million events and is compared with 7 different models. The data spans six distinct task domains: Typing, Seizure, Listening (Speech), Motor Imagery, P300 and Artifacts, with varying subject counts and recording volumes.

compresses each input window into a fixed-size latent grid, letting one architecture cover events from milliseconds to minutes, and a transformer decoder performs direct set prediction on top of it. Across ten public datasets (1,154 participants, 2.35 M events, six task domains), DANCE reaches a mean event-based F1 three times higher than the strongest baseline, sets a new state of the art on seizure monitoring, and matches onset-informed BCI accuracy without onsets. In the process we find that three large-scale EEG foundation models are among the strongest baselines on sample-level metrics yet lose almost all of their advantage at the event level, which suggests that pretraining objectives built for window classification do not transfer to asynchronous detection.

2 Event Detection in Time Series

Formally, we consider a segment of multi-channel EEG recording $\mathbf{X} \in \mathbb{R}^{T \times C}$, where T denotes the number of time samples and C the number of channels. The temporal dimension is defined as $T = f_s \cdot W$, where f_s is the sampling frequency and W the segment duration.

Unlike window-based methods, which assume a single label for the entire window, our objective is to *detect events*. We aim to map the input segment $\mathbf{X}$ to the set of events it contains, $\mathcal{E} = \{e_i\}_{i=1}^{N}$. Each event is parameterized as a tuple $e_i = (b_i, c_i)$, where $b_i = (t_{\text{start}}, t_{\text{end}}) \in [0, 1]^2$ denotes the normalized temporal boundaries relative to the segment duration W, and $c_i \in \{0, \ldots, K\}$ denotes the class of the event, where K is the number of event classes and $c_i = 0$ corresponds to the background class. Class definitions vary across datasets (see Table B.1).

There are two ways to evaluate event predictions: using sample-based metrics or using event-based metrics.

Sample-based Metrics In many prior studies, the event detection paradigm consists in a timepoint-wise classification task, and the most commonly used metric is the sample-level macro F1 score, called $\mathbf{F1}_{\text{sample}}$ in this work. However, while this formulation aligns with the training objectives of dense predictors (e.g., U-Net), it is unsuitable for quantifying event detection performance. This metric fails to penalize fragmented predictions, meaning that a single event predicted as multiple

disjoint segments can yield a high $\mathbf{F1}_{\text{sample}}$ despite failing to capture the event as a coherent unit. We report this metric to enable fair comparisons with prior works.

Event-based Metric To assess the model's ability to *decode* neural events, we introduce an event-level F1 score, $\mathbf{F1}_{\text{event}}$, which simultaneously quantifies detection and classification performance. We define the set of ground-truth events as $\mathcal{E} = \{e_i\}_{i=1}^{N}$ and predicted events as $\mathcal{P} = \{p_j\}_{j=1}^{M}$. A prediction p_j is matched to a ground truth e_i using a sorted greedy matching strategy consistent with standard object-detection protocols (Lin et al., 2014). A match is considered a True Positive if and only if:

$$\text{IoU}(p_j, e_i) > 0.5 \quad \text{and} \quad \text{class}(p_j) = \text{class}(e_i) \tag{1}$$

where IoU denotes the Intersection over Union of the time intervals. Predictions are sorted by class confidence if the IoU with ground-truth exceeds the threshold. A match is accepted only if neither element has already been matched. Unmatched predictions are counted as false positives, and unmatched ground-truth events as false negatives. The final metric is the F1 score derived from these counts.

$\mathbf{F1}_{\text{event}}$ operates at the event level and is independent of window size. It reflects the number of correctly localized events across the dataset, jointly assessing detection, classification, and temporal precision, and rewarding only accurate matches to ground truth. It effectively penalizes fragmented or duplicated predictions. In this work, we explicitly target this metric, which better evaluates the method's usability.

3 Architecture

We introduce **DANCE**, an architecture providing an end-to-end alternative to onset-informed EEG decoding by directly framing neural activity as an event-based prediction problem. As illustrated in Figure 1A, the framework comprises three sequential modules: a convolutional backbone for local feature extraction, a Perceiver-based temporal resampling module for duration adaptation, and a transformer decoder for set-based event prediction.

CNN Backbone: Local Feature Extraction The first stage processes the raw EEG input $\mathbf{X} \in \mathbb{R}^{T \times C}$ to extract a sequence of local latent representations for each timestep. For this, we use a modified convolutional architecture based on the work of Défossez et al. (2023).

The first component is a spatial attention module that is aware of the positions of the sensors, which encodes the C channels to a hidden dimension $D = 270$. It is followed by a sequence of 5 convolutional blocks with a kernel size of 9, which keep the temporal and hidden dimensions unchanged but capture multi-scale temporal dynamics with a dilation factor that increases by a factor of 2 at each layer. Finally, we project the channel dimension from D down to $H = 128$. The output is a sequence of dense embeddings $\mathbf{Z}_{dense} \in \mathbb{R}^{T \times H}$.

Perceiver: Temporal Resampling A core challenge in processing heterogeneous neural events is the range of their timescales, spanning from millisecond-level phonemes for speech perception to seizure episodes lasting several minutes. To unify the framework across datasets, we employ a Perceiver module (Jaegle et al., 2021), which translates the sequence of input embeddings of variable length T to a sequence of latents of fixed length $N_{Q_{\text{time}}}$.

The Perceiver maps $\mathbf{Z}_{dense}$ to a latent space with the same hidden dimension, $\mathbf{Z}_{\text{resamp}} \in \mathbb{R}^{N_{Q_{\text{time}}} \times H}$, via cross-attention over a set of $N_{Q_{\text{time}}}$ learnable latent queries $\mathbf{Q}_{\text{time}} \in \mathbb{R}^{N_{Q_{\text{time}}} \times H}$, effectively acting as a learnable, non-linear downsampling operation. To preserve temporal order, we initialize $\mathbf{Q}_{\text{time}}$ using fixed sinusoidal positional encodings uniformly spaced across the time window.

In our implementation, we set the latent length to $N_{Q_{\text{time}}} = 256$ and use a Perceiver depth of 6, configured with 2 attention heads for both the cross-attention and latent self-attention modules. Using a smaller $N_{Q_{\text{time}}}$ (e.g., $N_{Q_{\text{time}}} = 128$) makes short events, e.g. phonemes, which are on the order of 0.1 s, impossible to discern from one another, while larger sequence lengths increase memory use quadratically for the downstream decoder. Across datasets, $N_{Q_{\text{time}}} = 256$ strikes the best trade-off between temporal resolution and efficiency.

Due to the use of a fixed-size Perceiver, the input segment duration must be adapted for each dataset to strike a balance between coverage and resolution. It should be long enough to include the target

events, yet short enough to allow accurate boundary regression for each event. Additional details on how to select the segment size are provided in Appendix A.3.

Decoder: Set Prediction The final stage decodes the Perceiver's output to generate the final event predictions. We employ a standard transformer decoder as described in Carion et al. (2020) with 4 layers and 4 attention heads. The decoder transforms a set of $N_{Q_{\text{event}}}$ object queries $\mathbf{Q}_{\text{event}}$ into a set of event predictions. Unlike the Perceiver queries, $\mathbf{Q}_{\text{event}}$ are initialized randomly. The inputs can be defined as $\mathbf{Z}_{\text{sparse}} \in \mathbb{R}^{N_{Q_{\text{event}}} \times H}$. After a linear readout, the decoder outputs a set of prediction vectors $\hat{\mathcal{Y}} = \{\hat{\mathbf{y}}_i\}_{i=1}^{N_{Q_{\text{event}}}}$. Each prediction $\hat{\mathbf{y}}_i = (\hat{b}_i, \hat{c}_i)$ lies in the space $[0,1]^2 \times \Delta^{K+1}$, concatenating the predicted temporal coordinates $\hat{b}_i$ and the class probability distribution $\hat{c}_i$.

We choose to predict event start and end times rather than center and duration, as this formulation consistently yields better performance. In particular, it naturally mitigates border effects when an event is truncated at the beginning or end of a window: the model can independently predict each boundary, allowing partial events to be handled correctly.

Additional details on preprocessing and training procedures to ensure reproducibility, as well as baseline and ablation models, are provided in Appendix B. The full model code, along with the training and evaluation pipelines, will be released upon publication.

4 Loss Formulation

We train the model end-to-end using a loss $\mathcal{L}$ that enforces accuracy both at the dense signal level and the sparse event level, bridged by a consistency regularizer:

$$\mathcal{L} = \mathcal{L}_{sparse} + \mathcal{L}_{dense} + \lambda_{cons}\mathcal{L}_{cons} \tag{2}$$

In our experiments, we balance the contributions of these terms by setting $\lambda_{cons} = 0.5$.

Set Prediction Loss ($\mathcal{L}_{sparse}$) To evaluate the set prediction, we first define the pairwise matching cost between a ground truth element $y_i = (b_i, c_i)$ and a prediction $\hat{y}_{\sigma(i)} = (\hat{b}_{\sigma(i)}, \hat{p}_{\sigma(i)})$ indexed by a permutation σ. Here, b and c represent the normalized temporal segments and class labels, respectively, while $\hat{p}$ is the output of the softmax of the classification head. The matching cost $\mathcal{L}_{match}$ is a linear combination of classification and localization errors. We use the cross-entropy loss $\mathcal{L}_{cls}$ and the IoU loss $\mathcal{L}_{iou}$ to define it:

$$\mathcal{L}_{match} = \lambda_{cls}\mathcal{L}_{cls}(c_i, \hat{p}_{\sigma(i)}) + \lambda_{iou}\mathcal{L}_{iou}(b_i, \hat{b}_{\sigma(i)}) \tag{3}$$

We set the balancing coefficients to $\lambda_{cls} = 1$ and $\lambda_{iou} = 5$. We include a sensitivity analysis of the loss weights in Figure A.2. The objective is to find a bipartite matching that minimizes the cost between the ground truth set $\mathcal{Y}$ and the predicted set $\hat{\mathcal{Y}}$. Consequently, the final set prediction loss $\mathcal{L}_{event}$ is defined as the minimum over the set of all possible permutations $\mathfrak{S}_{N_{Q_{\text{event}}}}$:

$$\mathcal{L}_{sparse} = \min_{\sigma \in \mathfrak{S}_{N_{Q_{\text{event}}}}} \sum_{i=1}^{N_{Q_{\text{event}}}} \mathcal{L}_{match}(y_i, \hat{y}_{\sigma(i)}) \tag{4}$$

This assignment problem is addressed using the Hungarian algorithm (Kuhn, 1955), ensuring that the loss is computed based on the optimal one-to-one correspondence between events and predictions. With this algorithm, the bipartite matching problem can be solved in $\mathcal{O}\left(N_{Q_{\text{event}}}^3\right)$.

Dense Auxiliary Loss ($\mathcal{L}_{dense}$) Dense supervision methods are typically easier to optimize and yield meaningful predictions earlier in the training process. To combine the strengths of dense and sparse paradigms, we additionally attach a linear probe to the output of the Perceiver $\mathbf{Z}_{resamp}$ to predict the class for every timestep, giving an auxiliary cross-entropy loss:

$$\mathcal{L}_{dense} = \frac{1}{N_{Q_{\text{time}}}} \sum_{t=1}^{N_{Q_{\text{time}}}} \mathcal{L}_{ce}(\hat{y}_{t,dense}, y_{t,true}) \tag{5}$$

Consistency Loss ($\mathcal{L}_{cons}$) To enforce coherence between sample-based and event-based predictions, we project the discrete event-level predictions from the Decoder onto the temporal axis to create a reconstructed dense probability map P_{event}. For each predicted event, its predicted class probability $\hat{p}$ is assigned uniformly onto the temporal interval defined by its predicted boundaries $\hat{b}$. When multiple events overlap in time, their class probabilities are summed and normalized at each timestep to form a valid probability distribution. The consistency loss is defined as the Kullback–Leibler (KL) divergence between the reconstructed event distribution P_{event} and the timestep-level probability distribution P_{dense}:

$$\mathcal{L}_{cons} = D_{KL}\left(P_{\text{dense}} \parallel P_{\text{event}}\right) \tag{6}$$

This auxiliary objective encourages event-level outputs to be coherent with the dense predictions. The complete projection procedure and additional details are provided in Appendix B.3.3.

5 Experimental Setup

We evaluate on ten public datasets covering six task domains and event durations from 0.08 s to 67.7 s (Figure 1C, Table 1). Speech (Broderick, SparrKULee) and Typing (Pinet) probe fine-grained, high-frequency events; the Brain Invaders P300 suite (BI2013, BI2014a, BI2014b) has many events per window, where we detect both target and non-target stimuli; Motor Imagery (BNCI2014, BNCI2015) provides scarcer, longer endogenous events; and the Temple University Artifact (TUAR) and Seizure (TUSZ) corpora contribute long, overlapping and clinically critical events. All datasets share one minimal preprocessing pipeline (Lévy et al., 2025): bandpass 0.1–100 Hz, resampling to 128 Hz, per-channel robust scaling and clamping to $[-16, 16]$. Splits are cross-subject throughout, except TUSZ, which has an official fixed test split.

We compare against six baselines: a Random model informed by dataset statistics (a strong baseline, since it samples from the real event distribution), a window-based model (ATCNet; Altaheri et al. 2023), a segmentation model (U-Sleep; Perslev et al. 2021), and three EEG foundation models (LaBraM (Jiang et al., 2024), CBraMod (Wang et al., 2025a), REVE (El Ouahidi et al., 2025)). Since most published models only report onset-informed performance, we adapted them to the continuous setting by replacing their final classifier with a dense prediction layer. Two ablations isolate our contributions: DANCE w/out Decoder removes the sparse view, and DANCE w/out Hybrid keeps the architecture but drops the dense and consistency terms. Details are in Appendices B.4.3 and B.4.4.

Table 1: Summary of experimental datasets. Mean values are displayed.

Task Domain	Dataset	# Events / Window	Event Duration (s)
Speech	Broderick (Broderick et al., 2018)	75.33 (Phonemes)	0.08
	SparrKULee (Accou et al., 2024)	19.5 (Words)	0.3
Typing	Pinet (Zhang et al., 2025a)	19.3	0.11
P300	BI2013 (Congedo et al., 2011)	60.1	1
	BI2014a (Korczowski et al., 2019a)	45.8	1
	BI2014b (Korczowski et al., 2019b)	9.9	1
Motor Imagery	BNCI2014 (Tangermann et al., 2012)	1.3	4
	BNCI2015 (Faller et al., 2012)	3.4	5
Artifacts	TUAR (3.0.1) (Hamid et al., 2020)	11.03	9.04
Seizure	TUSZ (2.0.3) (Shah et al., 2018)	0.25	67.7

6 Results

Event-level performance. DANCE significantly outperforms every baseline on the event-based metric, reaching $\mathbf{F1}_{\text{event}} = 0.397 \pm 0.042$ — a threefold improvement over the strongest baseline,

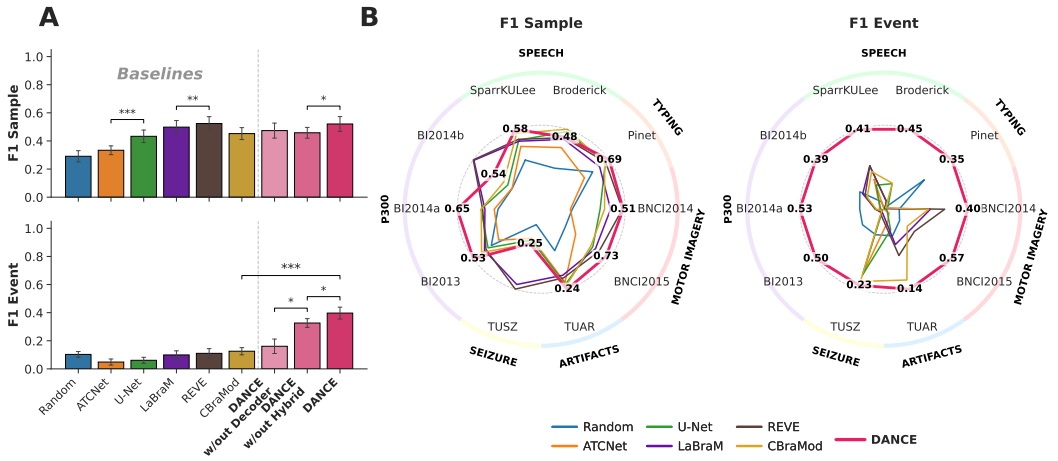

Figure 2: **DANCE yields a threefold improvement in event-based F1 over all baselines.** **A.** Performance averaged across subjects and datasets (Mean ± SEM) comparing baselines and ablations against the proposed DANCE model. Statistical significance via Linear Mixed-Effects Models (LMM; Appendix B.4.2) is indicated as follows: $^{*}p < 0.05$, $^{**}p < 0.01$, and $^{***}p < 0.001$. **B.** Normalized radar profiles showing relative averaged performance per subject across individual datasets. Axes are normalized to the best-performing method for each dataset. Numeric annotations indicate the absolute score of DANCE at each vertex, and the outer ring groups datasets by task category.

CBraMod (0.126; $p < 10^{-6}$, LMM; Figure 2A). On the sample-based metric the picture is very different: DANCE reaches $\mathbf{F1}_{\text{sample}} = 0.521 \pm 0.052$, statistically tied with REVE and LaBraM and only significantly above CBraMod ($p = 0.006$). A model can therefore score well on sample-based metrics while failing to produce usable events, which is precisely what the event-level metric exposes. The per-dataset breakdown (Figure 2B) also shows that DANCE avoids the catastrophic failures of conventional attention or CNN approaches, such as ATCNet on BNCI2014 or U-Sleep on BI2014b.

Pretraining does not transfer to asynchronous detection. The three foundation models are the most competitive baselines on $\mathbf{F1}_{\text{sample}}$, yet they lose almost all of that advantage once predictions must be read out as discrete events. Their dense outputs are fragmented, and this is not a thresholding artefact: heuristic post-processing such as gap-filling at varying thresholds fails to recover $\mathbf{F1}_{\text{event}}$ (Appendix B.2). Large-scale pretraining aimed at window-level classification thus does not, on its own, provide the temporally resolved representations that event detection needs — an argument for including asynchronous detection in the evaluation suites used to judge whether pretraining helps.

Temporal precision. To interpret performance in physical time units, we also report a class-agnostic detection F1 ($F1_{\text{det}}$) and a median center-point error (Δ_{ctr}, in seconds; Figure A.1B, Table A.1). DANCE dominates the best baseline on *both* axes on all ten datasets, with median center localization staying within 25% of event duration in every case: 14 ms on 80 ms phonemes, 78 ms on 4-second motor-imagery trials, and 4.5 s on multi-minute seizures. The best baseline is 2 to 19× less precise on cognitive tasks and 3× on BCI motor imagery, and the recall gap widens on long events, where DANCE retrieves 10× more seizures on TUSZ.

Both the sparse view and the hybrid objective are necessary. Each block contributes a statistically significant gain on $\mathbf{F1}_{\text{event}}$. The dense-only DANCE w/out Decoder reaches 0.161 and is clearly outperformed by DANCE w/out Hybrid (0.326, $p = 0.015$), confirming the value of set-based event prediction over dense readout. Adding the hybrid objective improves performance further, to 0.398 ($p = 0.025$), so the modified learning objective is not cosmetic: a bare set loss on EEG is not enough.

Decoded events. Figure 3 contrasts DANCE with CBraMod and U-Net on three datasets chosen for their different signal characteristics and event durations. On a 350-second TUSZ window (left),

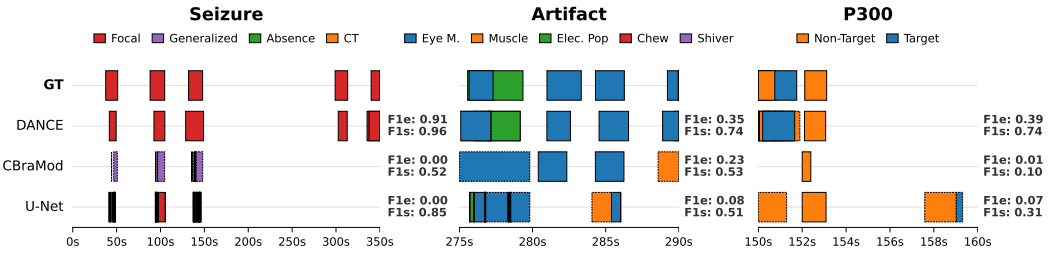

Figure 3: **DANCE yields qualitatively superior event predictions than baseline architectures.** Visualization of selected session segments comparing three architectures: U-Net, CBraMod, and our proposed model. Ground truth (GT) events are depicted alongside predictions from the respective models. Solid outlines indicate true positives, while dashed outlines denote false positives with respect to $\mathbf{F1}_{\text{event}}$. Colors represent distinct classes. $\mathbf{F1}_{\text{event}}$ (F1e) and $\mathbf{F1}_{\text{sample}}$ (F1s) are shown on the right for each portion.

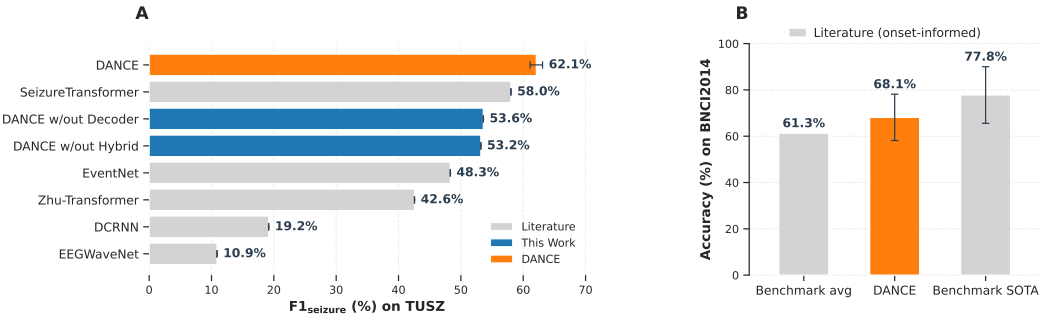

Figure 4: **DANCE establishes a new state-of-the-art for clinical seizure monitoring on the TUSZ dataset and matches onset-informed approaches on BNCI2014. A.** Benchmarking of DANCE against existing literature and ablation baselines using the $\mathbf{F1}_{\text{seizure}}$ metric. Scores are reported from Wu et al. (2025). DANCE is evaluated across 3 model seeds. TUSZ has a fixed test split. **B.** Intra-session performance on the BNCI2014 dataset for DANCE and the MOABB (Chevallier et al., 2024) benchmark. Standard deviation is reported.

DANCE detects and correctly classifies all five focal seizures, whereas the baselines find only the first three, misclassify them, miss the last two, and return fragmented segments; notably $\mathbf{F1}_{\text{sample}}$ fails to penalize U-Net here while $\mathbf{F1}_{\text{event}}$ does. On a 15-second TUAR segment (middle), DANCE separates distinct eye movements and identifies an electrode-pop artifact, while both baselines conflate artifact types and hallucinate a muscle artifact. On BI2014b (right), DANCE recovers the non-target events and approximately localizes the target, where CBraMod detects nothing and U-Net localizes reasonably but confuses classes. Across durations, inter-event gaps and signal complexities, DANCE produces temporally coherent proposals without any post-processing.

A new state of the art in clinical seizure monitoring. We benchmark DANCE against recent event-based seizure segmentation methods — SeizureTransformer (Wu et al., 2025), EventNet (Seeuws et al., 2024), ZhuTransformer (Zhu & Wang, 2023), DCRNN (Tang et al., 2022) and EEGWaveNet (Thuwajit et al., 2022) — on the fixed TUSZ test split, averaging three seeds (Figure 4A). DANCE reaches $\mathbf{F1}_{\text{seizure}}$ of 62.1% against 58.0% for the second-best method.[1] Raising the confidence threshold to 0.99 to suppress false alarms yields 1.88 false positives per $24\,\text{h}$ — a key clinical metric — while keeping $\mathbf{F1}_{\text{seizure}}$ above 58.0%, and all of this without post-processing.

Matching onset-informed pipelines without onsets. On BNCI2014 with the intra-session protocol of Chevallier et al. (2024), state-of-the-art onset-informed methods average 61.3% classification

[1]For seizures we use the binary $\mathbf{F1}_{\text{seizure}}$ to avoid over-reliance on the dominant background class, in line with recent recommendations (Dan et al., 2025).

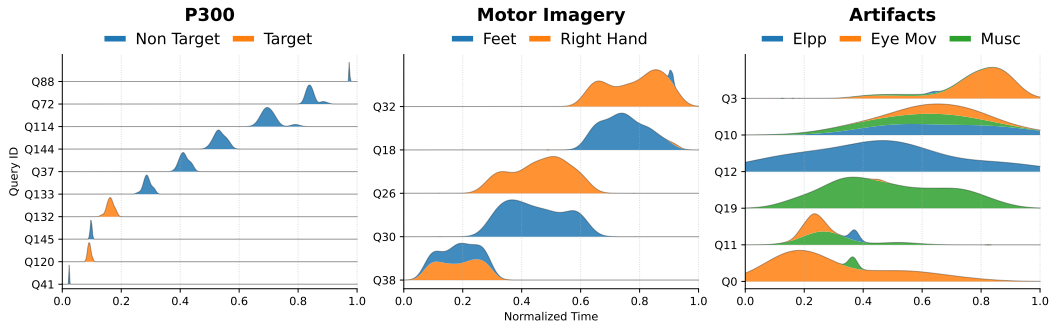

Figure 5: **DANCE queries dynamically specialize in specific temporal regions and event classes.** Visualization of the temporal distribution of predictions across selected queries. For each query row, the outer envelope corresponds to when the query activates. This is the total density of predictions. The colored bands indicate what it predicts. The stacked inner colored bands decompose this density by predicted class (most frequent at the base). A sharp, narrow envelope indicates temporal specialization, while a dominant color indicates class specialization.

accuracy, with the best model at 77.82%. DANCE, which must detect *and* classify, achieves 68.14% accuracy on temporally matched predictions, supported by a standard F1 of 96.57% over all predictions (Recall = 98.55%, Precision = 94.69%; Figure 4B). On highly structured datasets, competitive decoding therefore does not require onset information at all. Inspecting the learned queries shows why the formulation is flexible: queries specialize both temporally and by class, and adapt their sharpness to the structure of each dataset rather than following a fixed template (Figure 5).

Query Specialization By inspecting the learned queries, we can gain insights into how DANCE specializes across event classes and temporal localization. Figure 5 shows selected event queries across three datasets, where we visualize the distribution of predicted event centers: peaked distributions indicate temporal specialization, with queries consistently attending to narrow regions of the time window, while color composition reflects class specialization, with monochromatic distributions indicating queries that exclusively detect a specific class. The results for the P300 task (BI2013) are displayed on the left, with 10 of the 150 queries shown. We observe both temporal and class specialization, consistent with findings in computer vision models (Carion et al., 2020). Among queries sharing similar temporal localizations (e.g., Q145 and Q120), we observe specialization to different classes, indicating that queries effectively disentangle detection and classification. The middle panel shows Motor Imagery (BNCI2015), where temporal specialization remains clear but with broader distributions. Some queries are class-specific (Q30, Q18), while others attend to both classes (Q38). Finally, the right panel examines the Artifacts task: while some queries remain class-specialized (Q19, Q3), temporal distributions are more relaxed, with broad overlapping receptive fields rather than sharp localization, and some queries exhibit hybrid behavior by attending to multiple artifact types (Q10). Overall, these findings suggest that DANCE does not enforce a rigid template, but instead adapts its querying mechanism to the structure of each dataset.

7 Discussion

We introduced DANCE, an end-to-end model that jointly detects and classifies events in neural time series. That a single architecture performs well without onsets, across clinical, cognitive and BCI tasks, is a departure from current EEG practice, where a specialized onset-based model is built per task — SeizureTransformer (Wu et al., 2025) for seizures, ATCNet (Altaheri et al., 2023) for BCI. By framing event detection as set prediction and adapting it to noisy, heterogeneous EEG, we show that time-locked scenarios are not a prerequisite for high-performance decoding.

Limitations. For very short events such as phonemes or keystrokes, we restrict the task to voice detection: at this event density and SNR, even onset-based approaches struggle without broader context (Défossez et al., 2023), language priors (Zhang et al., 2025b) or higher-SNR modalities (Sivakumar et al., 2024; Willett et al., 2021). Second, although the architecture transfers across tasks

unchanged, we have not yet observed gains from training jointly across datasets, likely because of their heterogeneity and limited size; addressing this is the natural next step. The gap we report for CBraMod, REVE and LaBraM suggests that the opportunity is real but that it will require pretraining objectives designed for temporally resolved prediction rather than window classification.

AI USE STATEMENT

Generative AI tools assisted with software debugging, experiment orchestration and analysis, and manuscript editing and formatting. They were not used to generate or reformat datasets, formulate mathematical claims or proofs, or translate text. The authors reviewed and verified all AI-assisted code, analyses, claims, and text, and take responsibility for the final content of this work.

ETHICS STATEMENT

DANCE supports positive clinical applications, such as automated seizure monitoring, and asynchronous brain–computer interfaces for people with severe motor or communication impairments. Continuous neural decoding also raises risks of automation bias and mental-privacy violations: in clinical settings, the model should support rather than replace expert judgment, and deployment requires appropriate consent, security and oversight. This work uses only previously released public datasets and introduces no new data collection or immediate privacy risk.

REPRODUCIBILITY STATEMENT

The main text specifies the event formulation, metrics, architecture and loss (Sections 2 to 4). The appendices document preprocessing and dataset details (Appendix B), training hyperparameters and compute (Appendices B.3.4 and B.3.6), evaluation protocol and statistics (Appendices B.4 and B.4.2), and all baseline adaptations and ablations (Appendices B.4.3 and B.4.4).

REFERENCES

S Abirami, M Kathiravan, Rajamanickam Yuvaraj, Ramshekhar N Menon, John Thomas, PA Karthick, A Amalin Prince, Jac Fredo Agastinose Ronickom, et al. Automated multi-class seizure-type classification system using EEG signals and machine learning algorithms. *IEEE Access*, 12: 136524–136541, 2024.

Bernd Accou, Lies Bollens, Marlies Gillis, Wendy Verheijen, Hugo Van Hamme, and Tom Francart. Sparrkulee: A speech-evoked auditory response repository from ku leuven, containing the EEG of 85 participants. *Data*, 9(8):94, 2024.

Hamdi Altaheri, Ghulam Muhammad, and Mansour Alsulaiman. Physics-informed attention temporal convolutional network for EEG-based motor imagery classification. *IEEE Transactions on Industrial Informatics*, 19(2):2249–2258, 2023.

Hubert Banville, Yohann Benchetrit, Stéphane d'Ascoli, Jérémy Rapin, and Jean-Rémi King. Scaling laws for decoding images from brain activity. *preprint arXiv:2501.15322*, 2025.

Michael P Broderick, Andrew J Anderson, Giovanni M Di Liberto, Michael J Crosse, and Edmund C Lalor. Electrophysiological correlates of semantic dissimilarity reflect the comprehension of natural, narrative speech. *Current Biology*, 28(5):803–809, 2018.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision (ECCV)*. Springer, 2020.

Stanislas Chambon, Valentin Thorey, Pierrick J Arnal, Emmanuel Mignot, and Alexandre Gramfort. DOSED: A deep learning approach to detect multiple sleep micro-events in EEG signal. *Journal of neuroscience methods*, 321:64–78, 2019.

Félix Chavelli, Paul Boniol, and Michaël Thomazo. Toward interpretable evaluation measures for time series segmentation. *preprint arXiv:2510.23261*, 2025.

Sylvain Chevallier, Igor Carrara, Bruno Aristimunha, Pierre Guetschel, Sara Sedlar, Bruna Lopes, Sébastien Velut, Salim Khazem, and Thomas Moreau. The largest EEG-based BCI reproducibility study for open science: the moabb benchmark. *preprint arXiv:2404.15319*, 2024.

Marco Congedo, Matthieu Goyat, Nicolas Tarrin, Gelu Ionescu, Léo Varnet, Bertrand Rivet, Ronald Phlypo, Nisrine Jrad, Michaël Acquadro, and Christian Jutten. "Brain Invaders": a prototype of an open-source P300-based video game working with the openvibe platform. In *International Brain-Computer Interface Conference (BCI)*, 2011.

Alexander Craik, Yongtian He, and Jose L Contreras-Vidal. Deep learning for electroencephalogram (EEG) classification tasks: a review. *Journal of neural engineering*, 16(3):031001, 2019.

Jonathan Dan, Amirhossein Shahbazinia, Christodoulos Kechris, and David Atienza. Szcore as a benchmark: report from the seizure detection challenge at the 2025 AI in epilepsy and neurological disorders conference. *preprint arXiv:2505.18191*, 2025.

Alexandre Défossez, Charlotte Caucheteux, Jérémy Rapin, Ori Kabeli, and Jean-Rémi King. Decoding speech perception from non-invasive brain recordings. *Nature Machine Intelligence*, 5(10): 1097–1107, 2023.

Yassine El Ouahidi, Jonathan Lys, Philipp Thölke, Nicolas Farrugia, Bastien Pasdeloup, Vincent Gripon, Karim Jerbi, and Giulia Lioi. Reve: A foundation model for EEG – adapting to any setup with large-scale pretraining on 25,000 subjects. In *Advances in Neural Information Processing Systems*, 2025.

Josef Faller, Carmen Vidaurre, Teodoro Solis-Escalante, Christa Neuper, and Reinhold Scherer. Autocalibration and recurrent adaptation: Towards a plug and play online ERD-BCI. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 20(3):313–319, 2012.

Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

Ahmed Hamid, Katherine Gagliano, Safwanur Rahman, Nikita Tulin, Vincent Tchiong, Iyad Obeid, and Joseph Picone. The temple university artifact corpus: An annotated corpus of EEG artifacts. In *Signal Processing in Medicine and Biology Symposium (SPMB)*. IEEE, 2020.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *preprint arXiv:1503.02531*, 2015.

Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *International Conference on Machine Learning (ICML)*. PMLR, 2021.

Weibang Jiang, Liming Zhao, and Bao liang Lu. Large brain model for learning generic representations with tremendous EEG data in BCI. In *International Conference on Learning Representations (ICLR)*, 2024.

Louis Korczowski, Ekaterina Ostaschenko, Anton Andreev, Grégoire Cattan, Pedro Luiz Coelho Rodrigues, Violette Gautheret, and Marco Congedo. Brain Invaders calibration-less P300-based BCI using dry EEG electrodes dataset (bi2014a). Technical report, GIPSA-lab, 2019a.

Louis Korczowski, Ekaterina Ostaschenko, Anton Andreev, Grégoire Cattan, Pedro Luiz Coelho Rodrigues, Violette Gautheret, and Marco Congedo. Brain Invaders solo versus collaboration: Multi-user P300-based brain-computer interface dataset (bi2014b). Technical report, GIPSA-lab, 2019b.

H. W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955.

Vernon J Lawhern, Amelia J Solon, Nicholas R Waytowich, Stephen M Gordon, Chou P Hung, and Brent J Lance. EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces. *Journal of Neural Engineering (JNE)*, 15(5):056013, 2018.

Salomé Le Franc, Gabriela Herrera Altamira, Maud Guillen, Simon Butet, Stéphanie Fleck, Anatole Lécuyer, Laurent Bougrain, and Isabelle Bonan. Toward an adapted neurofeedback for post-stroke motor rehabilitation: state of the art and perspectives. *Frontiers in Human Neuroscience*, 16: 917909, 2022.

Jarod Lévy, Mingfang Zhang, Svetlana Pinet, Jérémy Rapin, Hubert Banville, Stéphane d'Ascoli, and Jean-Rémi King. Brain-to-text decoding: A non-invasive approach via typing. *preprint arXiv:2502.17480*, 2025.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*. Springer, 2014.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

Lena Maier-Hein, Annika Reinke, Patrick Godau, Minu D Tizabi, Florian Buettner, Evangelia Christodoulou, Ben Glocker, Fabian Isensee, Jens Kleesiek, Michal Kozubek, et al. Metrics reloaded: recommendations for image analysis validation. *Nature methods*, 21(2):195–212, 2024.

Joseph N Mak and Jonathan R Wolpaw. Clinical applications of brain-computer interfaces: current state and future prospects. *IEEE reviews in biomedical engineering*, 2:187–199, 2009.

Mathias Perslev, Sune Darkner, Lykke Kempfner, Miki Nikolic, Poul Jørgen Jennum, and Christian Igel. U-sleep: resilient high-frequency sleep staging. *NPJ digital medicine*, 4(1):72, 2021.

Joseph Redmon, S Divvala, R Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2015.

Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 658–666, 2019.

Robin Tibor Schirrmeister, Jost Tobias Springenberg, Lukas Dominique Josef Fiederer, Martin Glasstetter, Katharina Eggensperger, Michael Tangermann, Frank Hutter, Wolfram Burgard, and Tonio Ball. Deep learning with convolutional neural networks for EEG decoding and visualization. *Human brain mapping*, 38(11):5391–5420, 2017.

Skipper Seabold, Josef Perktold, et al. Statsmodels: econometric and statistical modeling with python. *SciPy*, 7(1):92–96, 2010.

Nick Seeuws, Maarten De Vos, and Alexander Bertrand. Avoiding post-processing with event-based detection in biomedical signals. *IEEE Transactions on Biomedical Engineering*, 71(8):2442–2453, 2024.

Vinit Shah, Eva Von Weltin, Silvia Lopez, James Riley McHugh, Lillian Veloso, Meysam Golmohammadi, Iyad Obeid, and Joseph Picone. The temple university hospital seizure detection corpus. *Frontiers in neuroinformatics*, 12:83, 2018.

Viswanath Sivakumar, Jeffrey Seely, Alan Du, Sean Bittner, Adam Berenzweig, Anuoluwapo Bolarinwa, Alex Gramfort, and Michael Mandel. emg2qwerty: A large dataset with baselines for touch typing using surface electromyography. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Leslie N Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial intelligence and machine learning for multi-domain operations applications*. SPIE, 2019.

Siyi Tang, Jared A Dunnmon, Khaled Saab, Xuan Zhang, Qianying Huang, Florian Dubost, Daniel L Rubin, and Christopher Lee-Messer. Self-supervised graph neural networks for improved electroencephalographic seizure analysis. In *International Conference on Learning Representations (ICLR)*, 2022.

Michael Tangermann, Klaus-Robert Müller, Ad Aertsen, Niels Birbaumer, Christoph Braun, Clemens Brunner, Robert Leeb, Carsten Mehring, Kai J Miller, Gernot R Müller-Putz, et al. Review of the BCI competition iv. *Frontiers in neuroscience*, 6:55, 2012.

Punnawish Thuwajit, Phurin Rangpong, Phattarapong Sawangjai, Phairot Autthasan, Rattanaphon Chaisaen, Nannapas Banluesombatkul, Puttaranun Boonchit, Nattasate Tatsaringkansakul, Thapanun Sudhawiyangkul, and Theerawit Wilaiprasitporn. EEGWaveNet: Multiscale CNN-based spatiotemporal feature extraction for EEG seizure detection. *IEEE Transactions on Industrial Informatics*, 18(8):5547–5557, 2022.

Jiquan Wang, Sha Zhao, Zhiling Luo, Yangxuan Zhou, Haiteng Jiang, Shijian Li, Tao Li, and Gang Pan. CBramod: A criss-cross brain foundation model for EEG decoding. In *International Conference on Learning Representations (ICLR)*, 2025a.

Shuo Wang, Chunlong Xia, Feng Lv, et al. RT-DETRv3: Real-time end-to-end object detection with hierarchical dense positive supervision. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2025b.

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.

Lukas Wolf, Ard Kastrati, Martyna Beata Płomecka, Jie-Ming Li, Dustin Klebe, Alexander Veicht, Roger Wattenhofer, and Nicolas Langer. A deep learning approach for the segmentation of electroencephalography data in eye tracking applications. In *International Conference on Machine Learning (ICML)*. PMLR, 2022.

H. Wu, S. Li, and D. Wu. Motor imagery classification for asynchronous EEG-based brain–computer interfaces. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 32:527–536, 2024.

Kerui Wu, Ziyue Zhao, and Bülent Yener. Seizuretransformer: Scaling U-Net with transformer for simultaneous time-step level seizure detection from long EEG recordings. *preprint arXiv:2504.00336*, 2025.

Mingfang Zhang, Jarod Lévy, Stéphane d'Ascoli, Jérémy Rapin, F Alario, Pierre Bourdillon, Svetlana Pinet, Jean-Rémi King, et al. From thought to action: How a hierarchy of neural dynamics supports language production. *preprint arXiv:2502.07429*, 2025a.

Xun Zhang, Jiaxing Li, Ruijie Zhang, and Tao Liu. A brain-controlled and user-centered intelligent wheelchair: A feasibility study. *Sensors*, 24(10):3000, 2024.

Yizi Zhang, Linyang He, Chaofei Fan, Tingkai Liu, Han Yu, Trung Le, Jingyuan Li, Scott Linderman, Lea Duncker, Francis R Willett, et al. Decoding inner speech with an end-to-end brain-to-text neural interface. *arXiv preprint arXiv:2511.21740*, 2025b.

Yuanda Zhu and May D Wang. Automated seizure detection using transformer models on multichannel EEGs. In *EMBS International Conference on Biomedical and Health Informatics (BHI)*. IEEE, 2023.

Zhuofan Zong, Guanglu Song, and Yu Liu. DETRs with collaborative hybrid assignments training. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

A Complementary Analysis

A.1 Metric robustness and decoupled analysis

$F1_{\text{event}}$ groups three facets of a prediction: detection, classification, and temporal precision. We decouple it with two complementary physical-unit metrics:

- $F1_{\text{det}}$: class-agnostic detection at IoU ≥ 0.5, isolating detection from classification.
- Δ_{ctr}: median center-point error of matched true positives, in seconds, isolating temporal precision from detection and classification.

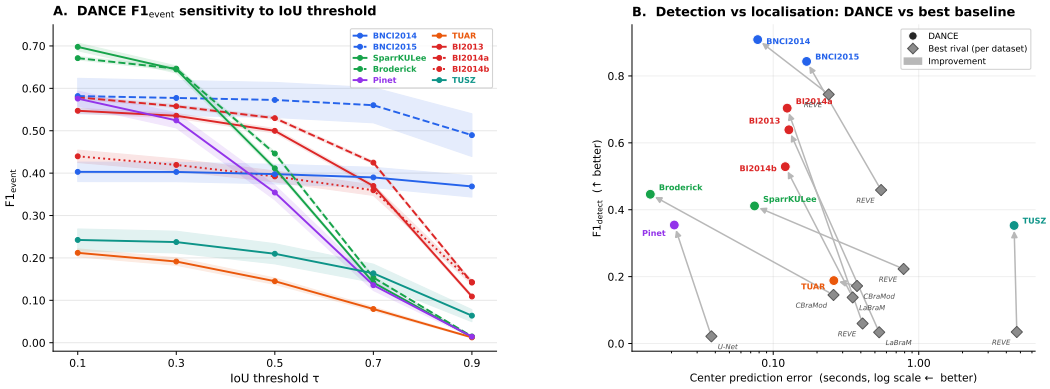

Figure A.1: **A.** $F1_{\text{event}}$ of DANCE as a function of the IoU acceptance threshold τ, one curve per dataset (coloured by task domain). **B.** Detection quality ($F1_{\text{det}}$, class-agnostic, ↑ better) versus temporal precision (Δ_{ctr}, center-point error in seconds, log scale, ← better) for DANCE (coloured) against the best non-ablation rival (grey). If the arrow points to the upper left, it means DANCE dominates on both axes. The arrow runs from the best baseline to the DANCE model.

Panel A shows that $F1_{\text{event}}$ decays monotonically and smoothly with τ on all ten datasets, with no cliff between the $\tau = 0.5$ default and the stricter $\tau = 0.9$ regime: the chosen operating point is not a knife-edge.

Panel B and Table A.1 report the physical-unit picture per dataset: DANCE dominates the best rival on both axes on *every one of the 10 datasets*. In absolute terms, Δ_{ctr} is 14–74 ms on speech (Broderick, SparrKULee), 21 ms on typing (Pinet), $\sim$ 125 ms across the three P300 datasets, 78–170 ms on BCI motor-imagery, 261 ms on TUAR artifacts, and 4.5 s on TUSZ seizures. Relative to event duration, this amounts to 17–25% on sub-second speech and typing events, $\sim$ 12% on P300 stimuli, and $\leq$ 7% on every other dataset. This is consistent with the main reported results. The predicted event center sits well inside the true event's temporal support, which is the precision required to time-lock a prediction to the underlying neural response for downstream analysis (ERP averaging, BCI feedback, clinical onset tagging). $F1_{\text{det}}$ is meanwhile $3\times$ to $19\times$ higher than the best rival on $6/10$ datasets (e.g. BI2013: 0.64 vs. 0.03 for LaBraM), and exceeds it on every dataset, so this precision is achieved while retrieving strictly more events.

Table A.1: Decoupled detection and localization metrics. $F1_{detect}$ measures class-agnostic detection (IoU≥0.5). $F1_{event}$ adds class awareness. Δ_{ctr} is the median temporal error of true positives in seconds. Best rival is the highest-scoring non-ablation baseline per dataset.

		DANCE			**Best Rival**		
Dataset	Rival	$F1_{det}$	$F1_{evt}$	Δ_{ctr} (s)	$F1_{det}$	$F1_{evt}$	Δ_{ctr} (s)
BNCI2014	REVE	0.909	0.398	0.078	0.745	0.288	0.241
BNCI2015	REVE	0.843	0.573	0.170	0.459	0.257	0.551
SparrKULee	REVE	0.411	0.411	0.074	0.223	0.223	0.787
Broderick	CBraMod	0.446	0.446	0.014	0.145	0.145	0.260
Pinet	U-Net	0.354	0.354	0.021	0.021	0.021	0.038
TUAR	CBraMod	0.188	0.145	0.261	0.172	0.122	0.377
BI2013	LaBraM	0.639	0.500	0.128	0.034	0.027	0.535
BI2014a	REVE	0.704	0.530	0.125	0.060	0.055	0.411
BI2014b	LaBraM	0.529	0.392	0.121	0.138	0.122	0.352
TUSZ	REVE	0.353	0.210	4.522	0.034	0.020	4.720

A.2 Sensitivity to the DETR loss weights

The DETR component of the hybrid loss balances temporal localization (λ_{IoU}) and event classification (λ_{class}). We selected the default configuration $(\lambda_{\text{IoU}}, \lambda_{\text{class}}) = (5, 1)$ to bias optimisation toward precise temporal boundaries before refining the discrete classification of those bounded segments. To show that the model is robust to this choice, we re-trained DANCE on the four corners of the $(\lambda_{\text{IoU}}, \lambda_{\text{class}}) \in \{1, 5\}^2$ grid on three datasets (BI2014b, BNCI2014, Broderick), keeping every other hyperparameter fixed and using the same 5-fold cross-validation protocol as in the main experiments. Figure A.2 reports both $F1_{\text{sample}}$ and $F1_{\text{event}}$ averaged across the three datasets. The two metrics vary by less than 0.05 F1 across the swept range, confirming the robustness

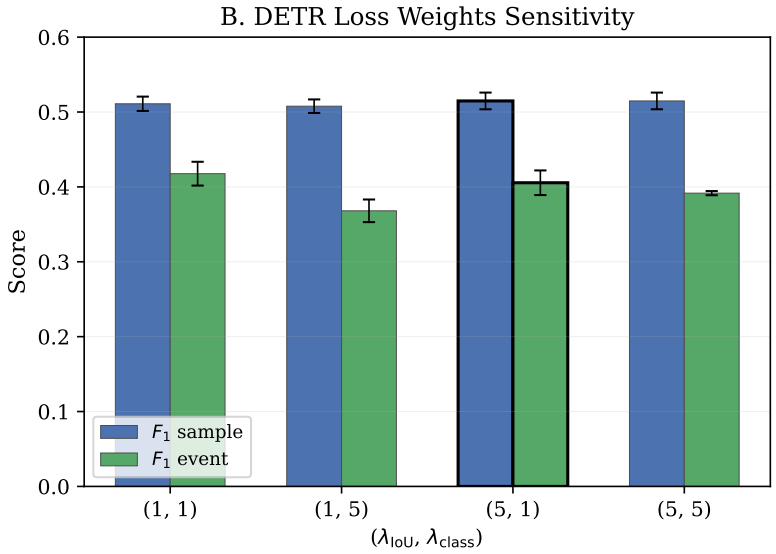

Figure A.2: **DANCE is robust to the DETR loss weights.** Mean $F1_{\text{sample}}$ (blue) and $F1_{\text{event}}$ (green) over BI2014b, BNCI2014 and Broderick (5-fold CV per dataset) for four configurations of the DETR localization and classification loss weights $(\lambda_{\text{IoU}}, \lambda_{\text{class}})$. Error bars are the SEM across datasets. The default configuration $(5, 1)$ used in every other experiment is highlighted with a thick border. Performance varies by less than 0.05 F1 across the swept range.

A.3 WINDOW SIZE IS DETERMINED BY ARCHITECTURAL CONSTRAINTS

The choice of the window size is fully determined by two architectural constraints:

- **Coverage (lower bound):** W must be long enough to fully contain the target events. For example, the 256 s window used on TUSZ is required because the average seizure duration is 67.7 s and routinely exceeds four minutes.
- **Resolution (upper bound):** DANCE resamples every window to a fixed latent of $N_{Q_{\text{time}}} = 256$ tokens (Appendix B.3). Accurate boundary regression requires each event to be represented by at least three latent tokens, yielding the rule of thumb

$$\frac{N_{Q_{\text{time}}}}{\bar{n}} = \frac{256}{\bar{n}} \geq 3\,,$$

  where $\bar{n}$ is the mean number of events per window.

Figure A.3 validates this design on the phoneme case, the dataset closest to the resolution bound: at $W = 8$ s (3.4 tokens/event) DANCE is at its optimum, while doubling W pushes $\bar{n}$ past the three-token cutoff and $F1_{\text{event}}$ collapses regardless of how queries are allocated. The fixed-$N_{Q_{\text{event}}}$ baseline falls to 18%, and scaling $N_{Q_{\text{event}}}$ in proportion to W drops further to $\sim 1\%$.

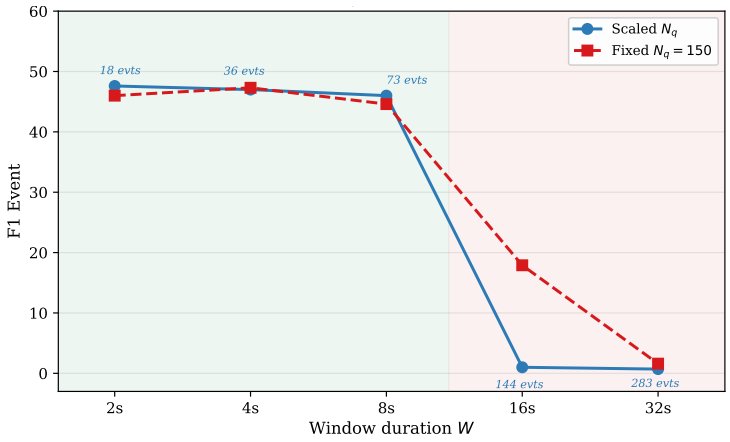

Figure A.3: **Window size is determined by the interplay of coverage and resolution.** Window-size sweep on the phoneme dataset. The solid blue curve scales the number of event queries with W; the dashed red curve keeps $N_{Q_{\text{event}}} = 150$ fixed. Blue annotations report the mean number of events per window at each W. The green band marks the regime where the Perceiver resolves each event with at least three latent tokens ($256/\bar{n} \geq 3$); the red band shows where resolution drops below this threshold and $F1_{\text{event}}$ collapses *regardless* of $N_{Q_{\text{event}}}$. $W = 8$ s, used in the main experiments, is the largest window that remains inside the green regime for this dataset.

Table A.2 shows that every other dataset sits comfortably inside the safe regime, with Tokens/Event ranging from 4.3 on BI2013 up to 1024 on TUSZ. This gives a clear heuristic on how to find the grid of valid $(W, N_{Q_{\text{event}}})$.

A.4 QUALITATIVE COMPARISON ON MOTOR-IMAGERY DATASETS

Figure A.4 shows decoded timelines of DANCE, CBraMod and U-Net on a single window of each motor-imagery dataset. DANCE preserves the coherent block structure of the ground truth, while CBraMod (foundation backbone with a dense head) misclassifies events and U-Net (sequence-labeling) produces the fragmented predictions characteristic of dense classifiers: the two failure modes that end-to-end set-prediction is designed to avoid.

Table A.2: Window configuration and Perceiver resolution across all datasets. The Perceiver module resamples every window into 256 latent tokens regardless of duration. Tokens/Event is computed as $256/\bar{n}_{\text{events}}$ and must remain ≥ 3 for accurate boundary regression. Broderick (3.4) sits at the empirically validated threshold; all other datasets have substantial headroom.

Dataset	W (s)	Events / Win	Event Dur. (s)	Tokens / Event
Broderick	8	75.3	0.08	3.4
SparrKULee	8	19.5	0.30	13.1
Pinet	8	19.3	0.11	13.3
BI2013	32	60.1	1.00	4.3
BI2014a	32	45.8	1.00	5.6
BI2014b	8	9.9	1.00	25.9
BNCI2014	8	1.3	4.00	197
BNCI2015	32	3.4	5.00	74.4
TUAR	16	11.0	9.04	23.2
TUSZ	256	0.25	67.7	1024

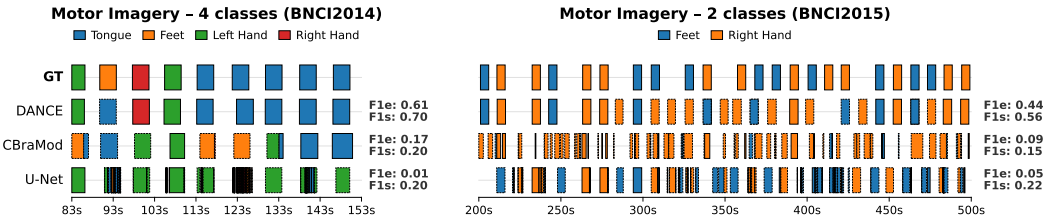

Figure A.4: **Qualitative timelines on motor-imagery datasets.** One row per method (top: ground truth; below: DANCE, CBraMod, U-Net). Coloured rectangles encode the predicted event class; solid borders mark predictions matching a ground-truth event (class and IoU ≥ 0.5), dashed borders mark mismatches. Left: BNCI2014 (4 classes), 83–153 s of one subject. Right: BNCI2015 (2 classes), 200–500 s of one subject. Per-row $F1_{\text{event}}$ (F1e) and $F1_{\text{sample}}$ (F1s) on this window are reported on the right.

B Details on Methods

B.1 Annotations, Label Standardization and Predictions

For eight of the ten datasets, annotations are used in their original format. However, the Temple University datasets (TUAR (Hamid et al., 2020) and TUSZ (Shah et al., 2018)) require specific handling as they provide channel-specific annotations. Since our framework targets global event detection, we project these annotations onto a single global timeline. Specifically, we merge simultaneous channel-level events, retaining a single event instance per start time per session to eliminate redundancy.

We also standardize the label spaces across tasks. For TUAR (artifacts), we define five distinct artifact classes: *muscle (musc), eye movement, electrode pop (elpp), chew,* and *shiver*, with composite events duplicated and reassigned to their corresponding classes. For TUSZ (seizures), annotations are mapped to four clinical seizure types: *focal, generalized, complex partial (CT),* and *absence*, following previous works (Abirami et al., 2024).

For the baseline models and the DANCE w/out Decoder variant, which rely on dense temporal predictions, we derive discrete events by extracting contiguous segments from the timestep prediction masks, resulting in a list of tuples defined by $(start, end, class)$. In contrast, for the DANCE w/out Hybrid ablation and the full DANCE architecture, the decoder head directly regresses these parameters, outputting explicit start coordinates, end coordinates, and class labels for each query within a window.

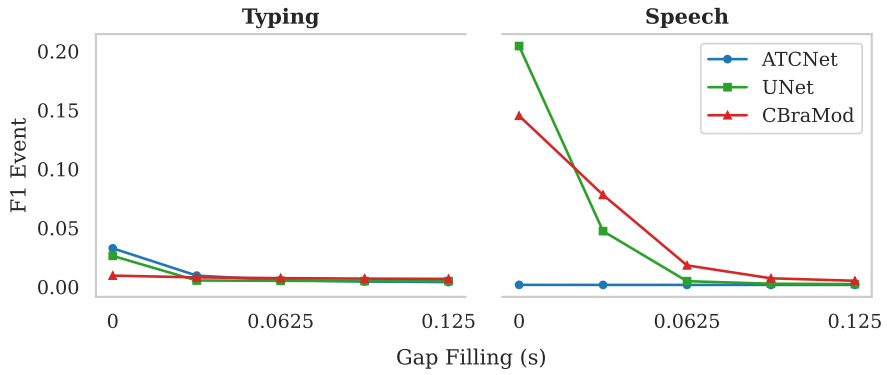

Figure B.1: **Temporal smoothing fails to improve event-based performance for baseline architectures.** The impact of the gap-filling parameter τ on the $\mathbf{F1}_{\text{event}}$ score for baseline methods (ATCNet, U-Net, CBraMod). For each $\tau \in [0, 0.125]$ s (x-axis), disjoint predictions of the same class are merged into a single continuous event if separated by a gap of $\leq \tau$. Results are shown for the Pinet (left) and Broderick (right) datasets.

B.2 Post-processing Evaluation for Baseline Models

Unlike standard object detection pipelines, our proposed method operates without non-maximum suppression (NMS) or duplicate-removal steps; predictions are directly interpreted as events.

To ensure a rigorous comparison, we investigated whether the baseline architectures (ATCNet, U-Net, CBraMod) would benefit from post-processing steps. As these models output dense, timestep-level probabilities, they are susceptible to prediction fragmentation, where continuous events are split into disjoint segments. We evaluated a gap-filling strategy where same-class segments separated by $\leq \tau$ are merged. The parameter range $\tau \in [0, 0.125]$ s was chosen to encompass durations up to the approximate mean length of typical speech and typing events. This range specifically targets the smoothing of short, spurious discontinuities.

As shown in Figure B.1, this heuristic did not yield consistent improvements in $\mathbf{F1}_{\text{event}}$. In many instances, increasing the temporal threshold τ degraded performance by erroneously merging distinct adjacent events.

While other post-processing steps exist, a central objective of this study is to validate the feasibility of a truly end-to-end pipeline that eliminates the dependency on manual intervention. The process of tuning post-processing stages introduces a selection of arbitrary thresholds or merging rules that adds significant complexity and risk of overfitting to specific datasets. Consequently, to ensure a strictly parameter-free evaluation and a transparent comparison, we report all baseline results based solely on raw model predictions.

B.3 Model Configuration, Losses and Training

B.3.1 Matching versus Evaluation

During training, the model optimization relies on the set-based loss without thresholding; instead, we utilize the Hungarian algorithm to enforce a global one-to-one assignment between the full set of model queries and the ground truth. Consequently, the majority of queries are matched to the background class to account for the sparsity of ground truth events relative to the query volume. Conversely, during evaluation, we select the class that yields the highest probability. A converged model effectively suppresses irrelevant queries by assigning them high probability in the background class. False positives are defined as non-background predictions that are incorrect, while false negatives correspond to ground-truth events unmatched by any valid foreground prediction.

B.3.2 Classification and IoU Losses

The classification loss $\mathcal{L}_{cls}$ is defined as the standard cross-entropy between the predicted probabilities $\hat{p}$ and the ground truth class c:

$$\mathcal{L}_{cls}(c, \hat{p}) = -\sum_{k} \mathbf{1}_{[c=k]} \log \hat{p}_k$$

The IoU loss $\mathcal{L}_{iou}$ for 1D temporal intervals with predicted (p_s, p_e) and target (t_s, t_e) is:

$$\text{IoU} = \frac{\max(0, \min(p_e, t_e) - \max(p_s, t_s))}{\max(10^{-6}, \max(p_e, t_e) - \min(p_s, t_s))}, \quad \mathcal{L}_{iou} = 1 - \text{IoU}.$$

Standard set-prediction architectures typically rely on Generalized IoU (Rezatofighi et al., 2019) to avoid vanishing gradients when predictions do not overlap with the ground truth. In our case, however, dense auxiliary supervision provides sufficient signal, making standard Intersection over Union adequate.

These losses are combined in the matching cost as:

$$\mathcal{L}_{match} = \lambda_{cls}\mathcal{L}_{cls} + \lambda_{iou}\mathcal{L}_{iou}$$

B.3.3 Consistency Loss Formulation

The consistency loss $\mathcal{L}_{cons}$ bridges the gap between the continuous dense stream predictions and event proposals. To compute it, we unfold the sparse event predictions into a dense map $P_{event} \in \mathbb{R}^{N_{time} \times K}$. This projection is used only as a probabilistic target and is differentiable solely with respect to the event class probabilities, not the predicted temporal boundaries. As such, the loss provides supervision to the semantic content of event predictions while leaving temporal localization unaffected.

First, for every timestep t in the downsampled sequence, we compute an unnormalized score $\tilde{P}_{event}(t, k)$ by aggregating the probability distributions of all predicted events covering that instant:

$$\tilde{P}_{event}(t, k) = \sum_{i=1}^{N_{Q_{\text{event}}}} \mathbf{1}_{[t \in \hat{b}_i]} \cdot \hat{p}_i(k) \tag{7}$$

where $\hat{p}_i(k)$ denotes the softmax probability of class k for the i-th query, and $\mathbf{1}[t \in \hat{b}_i]$ is the indicator function, equal to 1 if t falls within the predicted interval $\hat{b}_i$.

We then normalize these accumulated scores to obtain a valid probability distribution $P_{event}(t, k)$, introducing a small constant ϵ for numerical stability in regions devoid of event predictions:

$$P_{event}(t, k) = \frac{\tilde{P}_{event}(t, k)}{\sum_{j=0}^{K} \tilde{P}_{event}(t, j) + \epsilon} \tag{8}$$

Finally, $\mathcal{L}_{cons}$ introduced in Section 4 applies a Kullback–Leibler (KL) divergence loss between the dense temporal probability distribution P_{dense} and the unrolled event-derived probability map P_{event}.

We investigated replacing P_{dense} with the dense ground-truth labels and further evaluated a fully differentiable formulation of the consistency loss designed to propagate gradients also to the localization components. However, neither configuration outperformed the alignment of the P_{dense} and P_{event} probability distributions. This can suggest that the consistency term may act as a form of auto-distillation (Hinton et al., 2015), where the divergence between predictions provides a richer regularization than hard labels alone.

B.3.4 General Implementation Details

Table B.1 summarizes the dataset-specific hyperparameters. Across datasets, only three parameters vary: the window size, the number of queries and the batch size. We have defined in Appendix A.3

Table B.1: Additional information on the experimental datasets.

Task Domain	Dataset	Classes (K)	Window (s)	Queries
Motor Imagery	BNCI2014	L. Hand, R. Hand, Tongue, Feet	8	5
	BNCI2015	R. Hand, Feet	32	40
Speech	Broderick	Phonemes (Voice Detection)	8	150
	SparrKULee	Words (Voice Detection)	8	50
P300	BI2013	Target vs. Non Target	32	150
	BI2014a	Target vs. Non Target	32	150
	BI2014b	Target vs. Non Target	8	150
Seizure	TUSZ (2.0.3)	CT, Focal, Generalized, Absence	256	10
Artifacts	TUAR	Chew, Elpp, Eye Mov, Musc, Shiver	16	25
Typing	Pinet	Key Press	8	100

how to select the range for the window size. The number of queries must exceed the maximum number of events per window to guarantee full detection. Because excess queries simply predict "no-object" and linearly increase the bipartite matching cost during training, we set this parameter to a moderate multiple of the expected event count to balance coverage and efficiency. The batch size is chosen to maximize GPU utilization depending on the selected window size.

To ensure convergence and prevent overfitting, the convolutional backbone employs Batch Normalization coupled with GeLU activations for non-linearity. Regularization is managed via a dropout schedule, where a probability of $p = 0.1$ is applied to the raw input, while a rate of $p = 0.2$ is used within the convolutional blocks.

The complete architecture composed of the convolutional backbone, Perceiver, and the decoder is trained end-to-end. The model contains approximately 31 M parameters (27 M allocated to the encoder/perceiver and 4 M to the decoder). We use the AdamW optimizer (Loshchilov & Hutter, 2019). Unless otherwise specified, we train for 100 epochs with early stopping (patience = 20). We employ a OneCycle learning rate scheduler (Smith & Topin, 2019), with a warm-up phase covering the first 10% of training. The maximum learning rate is set to 5×10^{-5} for all datasets with the exception of the TUAR dataset, which required a higher rate of 10^{-4} to ensure convergence with our method.

$\mathbf{F1}_{\text{sample}}$ was computed using the `MultilabelF1Score` implementation from `torchmetrics`, while $\mathbf{F1}_{\text{seizure}}$ was computed using `BinaryF1Score`.

B.3.5 Seizure-Specific Optimization

Seizure detection presents unique challenges due to class imbalance; there is approximately 20 times more background than seizure activity in this dataset. To address this and inspired by Wu et al. (2025), we employ a weighted sampling strategy based on window occupancy. Let $o_i \in [0, 1]$ be the percentage of the window i containing seizure activity. The sampling weight w_i is calculated as:

$$w_i = \alpha + (1 - \alpha) \cdot \min(1, \beta \cdot o_i) \tag{9}$$

where $\alpha = 0.01$ ensures a minimum probability for background windows, and $\beta = 5$ boosts the sampling of sparse seizure events. We also introduce a 50% overlap between consecutive windows at training time to ensure consistent supervision for events spanning window borders. The model is trained for 25 epochs without early stopping. Importantly, with a duration = 256 s, the final window of each session is padded to the full window size, ensuring that inference can be performed over the entire test set. In Figure 4, we report the averaged performance across 3 model seeds on the fixed test set to account for variance between runs.

B.3.6 COMPUTE RESOURCES

DANCE has an average runtime of 8.03 h (training and evaluation) and an inference latency of 34.5 ms per single-window input. While higher than CBraMod (5.56 h, 28 ms), DANCE provides more robust performance and maintains the potential for real-time decoding with a limited increase in inference time.

Table B.2: Average runtime and inference latency across all datasets for each method.

	ATCNet	U-Net	CBraMod	DANCE w/out Decoder	DANCE w/out Hybrid	DANCE
Runtime (h)	1.18	1.14	5.56	3.51	5.85	8.03
Latency (ms)	2.35	4.65	28.02	34.81	34.03	34.53

All models were trained individually for each dataset on a single NVIDIA V100 GPU, except for the seizure detection task. Due to its larger scale, this dataset was trained using distributed data parallelism across 8 GPUs. Table B.2 reports the computational performance averaged across the 10 datasets: *runtime* corresponds to the total pipeline duration (training and evaluation), while *latency* represents the time required to process a single-window input during inference.

It should be noted that the metrics reported above reflect only the final, converged evaluation runs. The complete lifecycle of this research project required more computational resources. Developing the DANCE architecture, conducting hyperparameter sweeps to find a robust configuration across ten highly heterogeneous datasets, and evaluating preliminary designs that were ultimately discarded accounted for the majority of the total computational budget.

B.4 EVALUATION

B.4.1 SPLITTING AND AGGREGATION

We adopt a subject-independent evaluation protocol to ensure the generalization capability of the models.

- **General Protocol:** For most datasets, we employ a 5-fold cross-validation scheme grouped by subject. This approach ensures a complete partition of the dataset, effectively implementing a leave-N-subjects-out evaluation where each subject is tested exactly once. Metrics are first computed individually for each subject. Consequently, the results reported are the means across all subjects (Figure 2).
- **Seizure Detection:** We adhere to the official train/test split defined by the TUSZ dataset. Performance is evaluated on the entire held-out test set.

Regarding the highly structured protocols of the BNCI datasets, we ensured that the effective batch size defined as the product of batch size (BS) and window length (W) remains negligible relative to the total session duration. It is restricted to 64 s for BNCI2015 ($BS = 2, W = 32$ s) and 32 s for BNCI2014 ($BS = 4, W = 8$ s), very far from the mean session durations of 35.76 min and 6.45 min, respectively. This gap mitigates the risk of the model overfitting to macro-scale protocol regularities or memorizing absolute temporal positions within a session.

B.4.2 STATISTICAL ANALYSIS

We assessed statistical significance using Linear Mixed-Effects Models (LMMs), with subjects and datasets modeled as random effects to account for repeated measurements and domain-specific variability. This approach ensures that statistical significance reflects consistent effects across subjects and domains, rather than being driven by sample size imbalances or over-representation of specific datasets. Linear mixed-effects models were implemented using the `statsmodels` Python library (Seabold et al., 2010).

B.4.3 BASELINES AND ABLATIONS

We benchmark our approach against a diverse set of methods ranging from random baselines to state-of-the-art deep learning architectures.

Note on Architecture Selection. We exclude standard classification models tailored for single-window inputs, such as EEGNet (Lawhern et al., 2018). Adapting such architectures for temporal detection requires a sliding-window approach; however, a stride-1 evaluation leads to computationally intractable inference times (> 24 hours) on large-scale datasets. Consequently, we restrict our scope to architectures that can be modified or are designed for dense prediction.

Random Baseline. To establish a non-trivial lower bound, we generate random predictions by sampling directly from the list of all ground truth events in the dataset. For each session, the number of events injected is determined by the mean event rate in the dataset, while their start times are assigned uniformly at random.

ATCNet (Altaheri et al., 2023). We use ATCNet, an attention-based convolutional network, adapted here for dense prediction tasks. Our implementation utilizes the `braindecode`[2] library (Schirrmeister et al., 2017). To enable dense sequence labeling, we modified the architecture to preserve the input's temporal resolution. Specifically, we disabled temporal downsampling by setting the pooling parameters `conv_block_pool_size` to 1 and treated the input as a single continuous sequence by setting `n_windows` = 1. Furthermore, we retained the full output sequence of the Temporal Convolutional Network (TCN) rather than extracting only the final timestep. All other hyperparameters were maintained at their default values.

U-Sleep (Perslev et al., 2021). Given the dominance of U-Net architectures in the segmentation literature, we include the U-Sleep model (Perslev et al., 2021) as a strong baseline, utilizing the implementation provided by `braindecode`. To accommodate shorter input windows and prevent excessive feature collapse, we reduced the network depth from 12 to 8. Additionally, we modified the classification head by removing the adaptive pooling layer, ensuring the model produces a dense representation for every timestep.

LaBraM (Jiang et al., 2024). LaBraM is a transformer-based EEG foundation model pretrained on $\sim$2,500 hours of clinical and consumer-grade EEG. Pretraining follows a masked-token reconstruction objective: 1-second EEG patches are encoded by a vector-quantised tokenizer and the transformer is trained to recover the discrete tokens of randomly masked patches from their context, with shared per-channel embeddings allowing the same backbone to ingest heterogeneous electrode montages. We use the released pretrained weights and default configuration, and replace the final linear head with a multi-layer perceptron that produces a dense per-time-step class distribution.

REVE (El Ouahidi et al., 2025). REVE is a more recent EEG transformer pretrained on a $\sim$60,000-hour multi-source corpus mixing clinical, BCI and consumer-grade recordings. It's one order of magnitude larger than the corpora used by LaBraM or CBraMod. The architecture stacks 22 transformer blocks at hidden dimension 512, making it the largest of the three foundation backbones we evaluate. We follow the same protocol as for LaBraM and CBraMod.

Every baseline underwent a systematic search for optimal learning rates and batch sizes. For the dense baselines, we tested decision boundaries and gap-filling thresholds to maximize their temporal coherence. All models were required to converge across the 10-dataset benchmark using a unified configuration. This strict constraint prevents hyperparameter over-optimization.

B.4.4 Ablations

We conduct an ablation study to isolate the contributions of the proposed architecture and learning objectives:

1. **DANCE w/out Decoder:** This variant removes the DETR-style decoder and trains the backbone solely using the dense cross-entropy loss. This tests the efficacy of the encoder architecture in isolation.
2. **DANCE w/out Hybrid:** This variant includes the full architecture but removes the dense and consistency auxiliary losses from the supervision. This highlights the value of our

[2] https://braindecode.org/stable/index.html

hybrid training strategy in aligning the discrete (event) and continuous (signal) latent representations.