Impact of Temporal Window Shift on EEG-Based Machine Learning Models for Cognitive Fatigue Detection

Agnieszka Wosiak *, Michał Sumiński, Katarzyna Żykwińska

Posted Date: 2 September 2025

doi: 10.20944/preprints202509.0146.v1

Keywords: EEG; cognitive fatigue; window shift; overlapping windows; temporal segmentation; machine learning; evaluation protocol

 doi:10.20944/preprints202509.0146.v1

Article

Impact of Temporal Window Shift on EEG-Based Machine Learning Models for Cognitive Fatigue Detection

Agnieszka Wosiak *, Michał Sumiński and Katarzyna Żykwińska

Institute of Information Technology, Lodz University of Technology, al. Politechniki 8, 93-590 Łódź, Poland
* Correspondence: agnieszka.wosiak@p.lodz.pl

Abstract

In our study, we examine how the temporal window shift—the step between consecutive analysis windows—affects EEG-based cognitive fatigue detection while keeping the window length fixed. Using a reference workload dataset and a pipeline that includes preprocessing and feature extraction, we vary the shift to control segment overlap and, consequently, the number and independence of training samples. We evaluate six machine-learning models (Decision Tree, Random Forest, SVM, kNN, MLP, and a transformer). Across models, smaller shifts generally increase accuracy and F1 score, consistent with the larger sample count; however, they also reduce sample independence and can inflate performance if evaluation splits are not sufficiently stringent. Class-wise analyses reveal persistent confusion for the moderate-fatigue class, the severity of which depends on the chosen shift. We discuss the methodological trade-offs, provide practical recommendations for choosing and reporting shift parameters, and argue that temporal segmentation decisions should be treated as first-class design choices in EEG classification. Our findings highlight the need for transparent reporting of window length, shift/overlap, and subject-wise evaluation protocols to ensure reliable and reproducible results in cognitive fatigue detection.

Keywords: EEG; cognitive fatigue; window shift; overlapping windows; temporal segmentation; machine learning; evaluation protocol

1. Introduction

Reliable detection of cognitive fatigue from EEG is a long-standing goal in neuroergonomics and human factors, with immediate relevance to safety-critical contexts such as aviation, rail, and industrial process control. Despite clear progress in signal processing and machine learning for EEG, a persistent source of variability in reported performance stems from seemingly minor segmentation choices, especially the temporal window shift, defined as the step size between consecutive analysis windows. Because the shift simultaneously determines the degree of overlap between adjacent samples and the effective number and independence of training instances, it is a powerful—yet often under-documented—design lever that can inflate accuracy, change class confusions, and obscure what the models truly learn from brain signals. In this work, the term "cognitive fatigue" is used as a proxy operationalized via self-reported mental workload ratings from the STEW dataset.

In preliminary analyses conducted for this study on the STEW reference dataset [1,2], we observed that reducing the temporal window shift, while keeping the window length fixed, systematically improved performance across classical models and an illustrative lightweight transformer. As the shift decreased, the number of segments increased and classification metrics rose, but the confusion pattern also changed—especially for the moderate fatigue level, which remained the hardest class. These observations motivated the present study to isolate and quantify the impact of temporal window shift on EEG-based cognitive-fatigue classification under a controlled pipeline.

 doi:10.20944/preprints202509.0146.v1

Recent studies underscore the importance of this problem. In particular, reported advantages of certain models or feature sets can be confounded by evaluation protocols and segmentation choices. Brookshire et al. [3] showed that allowing segment-level leakage—meaning segments from the same subject appear in both training and test—can substantially inflate deep-learning accuracy in translational EEG. When the evaluation is enforced at the subject level, the reported performance drops markedly, revealing that the initial gains reflected the split rather than genuine generalization. Lee et al. [4] reached a similar conclusion in psychiatric EEG by contrasting subject-wise and trial-wise cross-validation under cropping-based augmentation. Their analysis demonstrated that inappropriate cross-validation produces optimistic metrics that do not carry over to unseen subjects, especially when overlapping segments from the same recording session are distributed across folds. Beyond leakage, Del Pup et al. [5] systematically varied data partitioning for cross-subject EEG deep learning and concluded that validation must be subject-based—and preferably nested—to avoid overestimation and to obtain stable performance estimates across architectures. Taken together, these findings indicate that segmentation and data-splitting choices are not bookkeeping details. They shape reported performance, influence class-wise behavior, and can change which models are ultimately judged superior under different protocols.

Several studies now examine overlap/shift directly. Falih et al. [6] studied sliding-window overlap ratios in EEG-based ASD diagnosis and found significant performance changes attributable purely to the degree of window overlap, which indicates that overlap (and, by extension, shift) is not a neutral parameter in the pipeline. In brain–computer interface and affective EEG, numerous pipelines employ overlapping windows as a matter of practice to densify samples and capture transitions, but only a fraction quantify the effect of step size itself on reported metrics or on class-wise behavior. Examples include overlapping-window designs for fatigue or affect models in which performance peaks at specific (length, step) combinations, underscoring that temporal sampling density functions as a practical control variable in both offline and online settings [7].

To avoid confounding effects of segmentation density with temporal content, we keep the window length fixed, because window duration itself is known to shape both discriminative power and class balance. Very short windows can miss slow trends relevant to fatigue, whereas very long windows can mix heterogeneous states within a single segment and blur class boundaries. Prior studies report performance changes driven by window length and task: accuracy varies with the chosen duration in epilepsy detection, with optimal ranges depending on the feature set and classifier [8]; in affective EEG, boredom classification improves at sub-second epochs, indicating that shorter windows emphasize transient dynamics [9]; and in cognitive workload classification, window length and step are jointly specified (e.g., 6-s windows with a 4-s step on STEW), further illustrating that step size is an explicit design choice with measurable effects [10]. By holding length constant, any change in the number of segments and in sample redundancy arises from the shift alone, which allows us to attribute downstream differences in accuracy and F1 to the step size rather than to altered temporal context.

The contribution of this work is to establish temporal shift as a first-order methodological factor in EEG-based classification of cognitive fatigue: small shifts densify observations and tend to improve metrics but also increase temporal redundancy and the risk of optimistic estimates unless evaluation rigor explicitly guards against leakage. Conversely, large shifts produce more independent segments but lower data volume, which can hinder learning—especially for models requiring many examples. Both regimes may be defensible depending on the scientific question (e.g., within-subject tracking vs. cross-subject generalization), but they must be reported and justified. Our objectives are therefore twofold: (i) to quantify the sensitivity of model performance and class-wise confusion to the shift parameter at fixed window length, and (ii) to distill practical recommendations for choosing and reporting window length, shift/overlap, and subject-wise evaluation protocols so that results are reproducible and comparable across labs. In line with these objectives, we explicitly state our evaluation assumptions (subject-wise splits), fix the feature-extraction pipeline across models, and vary only the step size to probe how segmentation density affects accuracy, F1, and class-specific errors.

 doi:10.20944/preprints202509.0146.v1

To make these aims concrete, we formulate the following research questions:

RQ1: Shift–performance sensitivity: When the window length is fixed, how does varying the temporal window shift influence overall accuracy and macro-F1 across the six models considered?

RQ2: Class-wise effects: How does the shift alter per-class precision and recall, and the structure of confusions, with particular attention to the moderate-fatigue level and across-participant variability?

RQ3: Data volume versus independence: How does the shift modulate the effective sample count and temporal redundancy, and to what extent do these factors mediate the observed performance differences?

RQ4: Model dependence: Are the effects of shift consistent across classical learners and a lightweight transformer, or do interactions between model capacity and shift emerge under the same segmentation and features?

These questions are addressed in Sections 4.1-4.4, respectively, and revisited in Section 4 to derive practical recommendations for reporting window length, shift/overlap, and evaluation protocols.

The remainder of the paper is organized as follows. Section 2 reviews the most relevant literature on EEG-based fatigue detection and classification techniques. Section 3 describes our materials and methods, beginning with the STEW dataset's key characteristics and then detailing preprocessing and the definition of temporal window shift and overlap. Section 4 presents the experimental results across six models and multiple shift values, including class-wise analyses, and discusses methodological implications, limitations, and recommendations. Finally, in Section 5, we summarize the conclusions, highlight practical implications for real-time fatigue monitoring, and outline future research.

2. Related Work

Research on EEG-based cognitive fatigue and workload spans engineered-feature pipelines with classical classifiers and end-to-end deep neural architectures. Early and still widely used approaches extract spectral and non-linear descriptors from fixed-length windows (e.g., band power, entropy, higher-order statistics) and apply k-nearest neighbours, support vector machines, decision trees, random forests, or shallow multilayer perceptrons. Such pipelines remain competitive on small-to-medium datasets and provide stable baselines when the main focus of the experiment is on methodological aspects rather than model architecture, especially when subject-wise evaluation is adopted and reporting is standardized [11–13]. Recent application-oriented reviews also emphasize practical constraints in deploying fatigue/workload systems and the continued relevance of lightweight models under limited data and operational requirements [14,15].

Over the last decade, deep learning has become a default family in many EEG applications. Compact convolutional networks capture spatial–spectral structure, while recurrent and hybrid CNN–LSTM models target temporal dependencies. Transformer-based models have more recently been explored for EEG either on raw time series or after time–frequency projection, often with convolutional front-ends that constrain local patterns before attention layers. Recent surveys and domain-specific reviews synthesize these developments and note both promise and sensitivity to data volume, label quality, and regularization choices [16,17]. Empirical studies illustrate the range of designs, from pure transformer classifiers and ensembles through CNN–transformer hybrids to attention mechanisms tailored to EEG structure [18–22]. Under strict cross-subject regimes and limited data, advantages over strong classical baselines are less universal and often depend on careful optimization, data augmentation, or adaptation strategies [23].

Beyond model families, reported performance in cognitive fatigue and workload studies reflects the chosen generalization regime. Within-subject designs typically yield higher scores and are common in adaptive or online scenarios, whereas cross-subject designs probe robustness to new individuals and provide a closer proxy for deployment. Recent work on reproducible workload classification highlights

 doi:10.20944/preprints202509.0146.v1

the importance of explicitly stating the evaluation regime, repetitions/seeds, and consistent metrics, including macro-F1 and class-wise precision/recall to expose behaviour on minority or borderline classes [13]. Survey articles likewise point to variability arising from differences in preprocessing, feature sets, and split strategies, reinforcing the need for transparent methodological reporting [11].

Window-based segmentation remains a shared prerequisite across most pipelines, irrespective of model capacity. Reported window lengths for cognitive workload and vigilance vary from sub-second to several seconds depending on the feature set and temporal granularity targeted by the task; step sizes are typically chosen to balance temporal coverage with sample density, particularly in streaming or quasi-online settings. Nevertheless, while many papers disclose the window length, fewer provide a systematic account of how the step size is selected relative to study goals (within-subject tracking versus cross-subject generalization) and data constraints, which motivates treating the step as an explicit experimental factor and reporting it alongside other hyperparameters [11,13].

Finally, comparative studies that include both classical and deep learners suggest a pragmatic division of labour: when labelled data are limited and cross-subject robustness is the priority, well-designed feature pipelines with classical classifiers offer competitive and stable baselines; with larger, diverse datasets or effective augmentation, deep architectures—particularly CNNs and hybrids, and in some settings transformers—can deliver additional gains [16,17,23]. Positioning temporal window shift as the main controlled factor in this study leverages that landscape and keeps architectural choices secondary to the central question of segmentation and its impact on performance and class behaviour.

3. Materials and Methods

3.1. Dataset Characteristics

The study utilized the Simultaneous Task EEG Workload (STEW) dataset, an open-access resource designed for research on mental workload using electroencephalography (EEG) signals [1,2]. The release comprises EEG recordings from 48 participants (50 recruited; 2 excluded in the release), each undergoing a resting-state session with eyes open and a SIMKAP multitasking session [1]. EEG was acquired with an Emotiv EPOC headset at 128 Hz (16-bit), using 14 electrodes positioned according to the international 10–20 system (AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4). The electrode montage is shown in Figure 1.

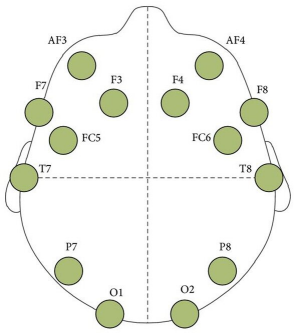

Figure 1. Electrode positions (10–20 system) used in STEW [1].

Each participant completed two sessions: a resting-state recording and the SIMKAP multitasking test, a standardized assessment of multitasking performance [1]. For both sessions, approximately 3 minutes of EEG were recorded; following the dataset protocol, the first and last 15 seconds are discarded, yielding 2.5 minutes of usable data per condition.

Following each session, participants self-reported perceived mental workload on a 9-point Likert scale [24], where 1 indicates minimal workload and 9 indicates maximal workload. Consistent with the dataset convention, ratings 1–3, 4–6, and 7–9 are mapped to *low*, *moderate*, and *high* workload, respectively, and serve as the three target classes in this study. Because labels are provided per recording segment, window-level samples inherit the segment label (the windowing procedure is detailed in Section 3.3).

 doi:10.20944/preprints202509.0146.v1

3.2. Processing Pipeline

The processing pipeline comprises four preprocessing stages (Figure 2): (i) band-pass filtering, (ii) artifact handling using Independent Component Analysis (ICA), (iii) window-level statistical feature extraction on the analysis windows defined in Section 3.3, and (iv) normalization aligned with the evaluation protocol. Model training is performed subsequently on the resulting feature vectors. Segmentation parameters (window length L and temporal shift S) are specified in Section 3.3.

Figure 2. Processing pipeline.

3.2.1. Bandpass Filtering

Raw EEG is band-pass filtered to retain task-relevant activity while suppressing slow drifts and high-frequency noise. Bandpass filtering selectively allows frequencies within a specified range to pass while weakening those outside the band. It may combine high-pass and low-pass filtering in a single operation, isolating desired components and suppressing unwanted noise and drift. Common design approaches include infinite impulse response (IIR) filters such as Butterworth and Chebyshev and Finite Impulse Response (FIR) filters, each with distinct frequency response and computational complexity trade-offs. Butterworth filters offer a maximally flat passband to preserve amplitude fidelity within the target range, avoiding ripples that could distort signal amplitude within targeted bands and making them popular in biomedical signal processing. Chebyshev filters achieve steeper roll-off at the expense of passband ripple, suitable when rapid attenuation of adjacent bands is required. FIR filters can be designed to show an exactly linear phase, ensuring no phase distortion of time-domain waveforms, although they typically require higher filter order and greater computation. Digital implementations often utilize zero-phase filtering, such as forward-reverse filtering, to avoid phase shifts that can distort event-related potentials [25].

In EEG applications, bandpass filtering isolates neural oscillations by removing artifacts from muscle activity, eye movements, and power-line interference. Proper cutoff selection depends on the signal characteristics and application needs, with typical EEG bandpass settings ranging from 0.5–1 Hz for high-pass to 30–70 Hz for low-pass filtering, often followed by notch filters at 50/60Hz [26].

Our study employed a fifth-order Butterworth bandpass filter with a 0.5 Hz low-cut and 50 Hz high-cut, implemented via zero-phase forward–reverse filtering. Setting the low cut at 0.5 Hz effectively removes slow, non-neural drifts (e.g., galvanic skin responses or electrode polarization) without distorting event-related potentials since higher cutoffs (> 0.5 Hz) have been shown to produce waveform distortions and attenuate critical low-frequency components. The high cut at 50 Hz excludes power-line interference (50 Hz in Europe) and other high-frequency artifacts while preserving beta and gamma rhythms up to ~ 45 Hz, which are implicated in attention and fatigue processes. A fifth-order design achieves a steep roll-off (≈ 30 dB/octave), balancing artifact suppression against minimal signal phase distortion. These settings match common configurations reported in recent cognitive workload and feature-extraction studies [10,12].

3.2.2. Artifact Handling Using Independent Component Analysis

Independent Component Analysis (ICA) decomposes multichannel signals into statistically independent components by maximizing non-Gaussianity (e.g., via kurtosis or negentropy). It is a blind source separation technique designed to recover individual sources from observed linear mixtures without prior knowledge of the mixing process. By leveraging higher-order statistics, ICA aims to recover latent sources with minimal assumptions about their temporal structure [27].

 doi:10.20944/preprints202509.0146.v1

In EEG signal processing, ICA is widely employed to isolate and remove artifacts without explicit reference channels. Popular implementations such as FastICA and Infomax use fixed-point iteration or information-maximization criteria to achieve stable convergence and reliable decompositions. After decomposition, components are inspected through their spatial topographies and temporal–spectral profiles to identify artifact-related sources (e.g., ocular components with frontal dominance and low-frequency power; myogenic components with broadband high-frequency content) for rejection or correction. Recent developments integrate ICA with machine-learning classifiers that label components (e.g., using annotated resources such as the EPIC EEG Independent Component dataset) or combine ICA with sequence models for automatic artifact estimation [28].

We applied FastICA with the number of components set equal to the number of EEG channels (14), ensuring a full-rank decomposition and maximal separation capacity. Components were marked for removal using a predefined decision rule based on (i) scalp maps, (ii) power spectral density, and (iii) time-course inspection for event-locked or burst-like artifacts. Signals were subsequently re-referenced to the common average to reduce bias from any single electrode. In line with recent evidence that ICA-based artifact removal does not universally improve downstream decoding, particularly in deep models or when artifacts are weak or nonstationary, we applied removals conservatively and only to components meeting strict criteria [29,30].

3.2.3. Wavelet-Based Feature Extraction

Wavelet transforms (discrete - DWT and continuous - CWT) decompose signals into scaled and shifted versions of a prototype function, enabling simultaneous time-frequency analysis of non-stationary data. Wavelets can capture transient features such as sudden shifts or oscillatory bursts by adapting time resolution at high frequencies and frequency resolution at low frequencies. In DWT, the signal undergoes successive high- and low-pass filtering with downsampling to produce detail (i.e., high-frequency) and approximation (i.e., low-frequency) coefficients at each level, facilitating multi-scale representation. CWTs compute coefficients over a continuum of scales and translations, yielding redundant, highly detailed time-frequency maps useful for visual analysis but at greater computational cost. Wavelet families (e.g., Daubechies, Symlet, or Coiflet) are selected based on properties like compact support, regularity, and vanishing moments, which influence the transform's sensitivity to features of interest [31].

In EEG preprocessing, wavelet transforms serve multiple purposes: noise and artifact reduction via thresholding of coefficients, extraction of band-specific power dynamics, and detection of event-related potentials or pathological spikes. Empirical wavelet transforms adapt filter shapes to the signal's spectral content, improving artifact segmentation in challenging contexts like motion artifacts or muscle noise [12,32,33].

3.2.4. Statistical Feature Extraction

We extracted a time-domain statistical feature set per channel that is suitable for cross-subject EEG classification and efficient enough for potential real-time use. Following established practice in recent EEG workload/fatigue pipelines, time-domain descriptors are used to capture waveform morphology and variability within each analysis window, offering robust baselines under limited data and heterogeneous subjects [11,12,32].

Specifically, each preprocessed EEG window was converted into a 13-dimensional statistical vector per channel comprising:

- *Central tendency and dispersion*: mean, standard deviation, variance.
- *Range and energy-related measures*: peak-to-peak amplitude (ptp), minimum, maximum, mean-square, root-mean-square (RMS).
- *Indices of extrema*: index of the minimum, index of the maximum (useful to indicate within-window timing of salient excursions).

 doi:10.20944/preprints202509.0146.v1

- *Shape descriptors*: skewness and kurtosis (higher-order moments summarizing asymmetry and tail heaviness).
- *Absolute successive differences*: a simple variability proxy sensitive to short-lived fluctuations within the window.

These statistics assume approximate stationarity at the window scale but are sensitive to transient dynamical changes in EEG that accompany shifts in perceived cognitive workload or fatigue. Their low computational cost and interpretability make them attractive for streaming or embedded settings, a point emphasized in recent surveys and methodological reviews [12,32]. In parallel to these time-domain descriptors, wavelet-based subband features (discrete wavelet transform with Daubechies-4 and a 5-level decomposition) were computed in a separate step (see Section 3.2, Wavelet Transformation), and the statistical set described here is therefore complementary.

Per-channel statistical vectors were concatenated across all 14 channels, forming a $13 \times 14 = 182$-dimensional statistical feature vector for each window. Concatenation preserves spatial detail while keeping the representation compact and consistent across subjects. To prevent information leakage, feature scaling was performed with z-scores fitted *exclusively* on the training portion of each subject-wise split (mean removal and variance scaling per feature) and then applied to the corresponding test portion. This leakage-aware normalization follows recent recommendations for reproducible EEG workload classification and ensures that reported performance reflects generalization rather than distributional hints from the test data [13]. No per-window label smoothing was used; windows inherit the segment-level labels defined in 3.1.

3.3. Windowing and Temporal Shift

We segment each continuous EEG recording into fixed-length windows and assign to every window the segment-level label (low, moderate, high) derived from the 9-point self-report (1–3, 4–6, 7–9; see Section 3.1). The window length is fixed at $L = 512$ samples (4 s at 128 Hz). We chose $L = 4$ s to stabilize the window-level statistics. This duration provides enough samples to yield reliable time-domain descriptors (e.g., RMS, kurtosis) and to span multiple cycles of theta/alpha rhythms relevant to workload, while remaining short enough to track within-session fluctuations without mixing heterogeneous states. Prior work shows that classification is sensitive to window duration and commonly adopts windows of a few seconds for cognitive workload. Our choice follows this practice and improves comparability with recent STEW-based studies [8–10].

Temporal shift S - the experimental factor - denotes the step between the start times of consecutive windows. The overlap ratio between adjacent windows is expressed by Eq. 1.

$$\text{overlap ratio} \;=\; 1 - \frac{S}{L}. \tag{1}$$

We evaluate $S \in \{32, 64, 128\}$ samples corresponding to 0.25 s, 0.50 s, and 1.00 s shifts, which by Eq. 1 implies overlap ratios of 93.75%, 87.50%, and 75.00%, respectively. This design reflects common practice in EEG classification where window length and step are specified jointly [10], and it allows us to examine how denser sampling (smaller S) affects both performance and class-wise behavior under subject-wise evaluation [3–5]. Prior work has shown that overlap can substantially alter reported metrics even when the model and features are unchanged [6].

For a recording of duration T seconds, the number of windows equals

$$N_{\text{win}}(T; L, S) = \left\lfloor \frac{T - \frac{L}{f_s}}{\frac{S}{f_s}} \right\rfloor + 1, \tag{2}$$

with $f_s = 128$ Hz. In STEW dataset, each condition lasts approximately $T \approx 150$ s [1,2]. Using Eq. 2, this yields $N_{\text{win}} = 585$ windows per recording for S=32 samples, 293 for S=64, and 147 for S=128. We process 90 subject–session recordings with segment-level labels distributed as: low n=42,

 doi:10.20944/preprints202509.0146.v1

moderate n=23, high n=25. Six recordings from the 96 available were excluded due to failed quality control (excessive artifacts) and/or missing workload ratings. Because T and L are fixed, per-class window counts equal $n \times N_{\text{win}}$ for a given S.

Table 1. Windows per recording and totals across N=90 recordings, for each shift S.

No of samples S (shift)	Shift (in s)	Overlap (%)	Windows per recording	Total windows
128	1.00	75.00	147	13 230
64	0.50	87.50	293	26 370
32	0.25	93.75	585	52 650

Table 2. Windows per class (low, moderate, high) by shift S.

No of samples S (shift)	Low (N=42)	Moderate (N=23)	High (N=25)	Total (N=90)
128	6 174	3 381	3 675	13 230
64	12 306	6 739	7 325	26 370
32	24 570	13 455	14 625	52 650

Reducing S increases the number of windows available for training and testing, but it also increases temporal redundancy because adjacent windows share a large proportion of samples (75–94% here). This redundancy may induce autocorrelation between neighboring windows, so the additional examples are not independent observations. Throughout this study we use subject-wise evaluation, which prevents windows from the same subject appearing in both the training and the test sets. Even with that safeguard, within-subject clustering remains and can raise performance when S is small because the model sees many highly similar windows from the same recording. The implication is that improvements observed at small S should be interpreted as the effect of denser sampling rather than evidence that the underlying task became easier. This interpretation aligns with recent analyses showing that segmentation and splitting choices can inflate or reshape reported metrics if they are not specified carefully [3–5]. We therefore treat S as a tuned and reported design choice to make results reproducible and comparable across studies [6,10].

3.4. Classifier Models

We evaluate five classical machine learning models - decision tree (DT), random forest (RF), k-nearest neighbors (kNN), support vector machine (SVM), and multilayer perceptron (MLP) - and a transformer-based deep network. All the methods were described in subsequent sections.

3.4.1. Traditional Machine Learning Models

Decision Trees (DTs) are non-parametric models that recursively partition the feature space into homogenous subsets by selecting feature thresholds that maximize class purity, typically via Gini impurity or information gain. At each node, the best split is chosen to minimize impurity in child nodes, yielding an interpretable tree of decision rules. This model can capture complex nonlinear patterns without requiring feature scaling, but individual trees often overfit small datasets [34]. We used the Gini criterion and no maximum depth in our experiments, allowing the tree to grow fully to capture subtle fatigue-related distinctions in our 13-dimensional feature set. The unbounded configuration follows EEG studies showing that fully grown trees can excel when combined with ensemble methods [35].

Random Forest (RF) mitigates decision trees' variance by training an ensemble of trees on bootstrapped subsets of the data and random feature subsets at each split, then aggregating predictions via majority voting. Random forests have repeatedly achieved top performance in EEG-based fatigue and workload detection tasks [34]. We implemented a random forest classifier with 100 trees, Gini impurity, and the size of the subsets of features to consider when splitting a node equal to the square root of the total features.

 doi:10.20944/preprints202509.0146.v1

k-Nearest Neighbors (kNN) is an instance-based classifier that assigns a class to a new sample based on the majority label among its k closest neighbors in feature space, measured by Euclidean distance. kNN makes minimal assumptions about data distributions, but it can be sensitive to noise and irrelevant features. Prior EEG studies have shown that kNN with small k values tends to perform well on statistical feature vectors. Accordingly, we set k=5 and used Euclidean distance, following benchmarks demonstrating that this choice optimally balances bias and variance on similar datasets [36].

Support Vector Machines (SVMs) seek hyperplanes that maximize the margin between classes in a high-dimensional space, with kernel functions enabling nonlinear decision boundaries. The Radial Basis Function (RBF) kernel is particularly well-suited for EEG features, capturing complex patterns without explicit feature transformations. SVMs have achieved strong performance in workload and fatigue classification when tuned with appropriate regularization ('C') and kernel bandwidth ('gamma') parameters [10]. We used an RBF kernel with default settings (C=1.0, gamma='scale'), offering a robust starting point that balances margin maximization and computational tractability.

Multilayer Perceptron (MLP) networks are feed-forward neural models with one or more hidden layers, enabling the learning of complex, nonlinear mappings via backpropagation and gradient-based optimization. Despite their "black-box" nature, shallow MLPs (one hidden layer) have proven effective on engineered EEG features, providing a balance between model capacity and overfitting risk. We configured an MLP with a single hidden layer of 100 neurons, ReLU activation, and a softmax output layer. We trained MLP using the Adam optimizer (learning rate 0.001) for up to 200 epochs with early stopping (patience 20). This setup reflects established EEG benchmarks and leverages dropout (rate 0.5) to regularize the network [37].

3.4.2. Transformer Model

Transformer networks employ self-attention mechanisms to model pairwise interactions across sequence elements, capturing long-range dependencies more effectively than recurrent architectures. An encoder block comprises multi-head self-attention, residual connections, layer normalization, and position-wise feed-forward sublayers, enabling parallel processing of sequence tokens with minimal inductive bias. Recent reviews highlight transformers' ability to learn hierarchical representations from raw EEG data with reduced reliance on manual feature engineering [38].

In our implementation (Figure 3), the model begins with an embedding layer projecting the 13-dimensional feature vectors into a 32-dimensional space, followed by a multi-head attention module with four heads and key dimension 32, and a point-wise feed-forward network of width 64 with ReLU activations. Dropout (rate 0.1) is applied after attention and feed-forward layers to prevent overfitting. We trained the network using Adam (learning rate $1x10^{-3}$) and categorical cross-entropy loss for up to 100 epochs with early stopping (patience 10), following hyperparameter settings validated in EEG transformer studies [39]. Our architecture balances expressivity and computational cost, aiming to leverage global context modeling for fatigue detection while mitigating data scarcity challenges.

 doi:10.20944/preprints202509.0146.v1

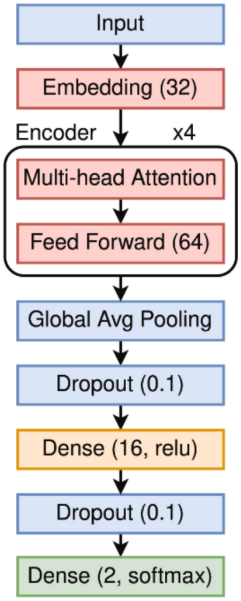

Figure 3. The architecture of the proposed transformer neural network model.

3.5. Evaluation

All models were trained and evaluated using a hold-out protocol. The data were randomly split into training (75%) and test (25%) sets using a subject-wise splitting protocol with a fixed random seed to ensure that data from each participant appears exclusively in one set, preventing data leakage. Preprocessing and feature scaling followed the pipeline in Sections 3.2–3.3. In particular, z-score parameters were estimated on the training portion and then applied to the test portion. For each temporal shift $S \in \{32, 64, 128\}$ and for each model, training was performed on the training set and all metrics were computed on the held-out test set.

We report standard multi-class metrics. Overall accuracy is the fraction of correctly classified windows in the test set:

$$\text{Accuracy} = \frac{\#\{\hat{y}_i = y_i\}}{N_{\text{test}}}. \tag{3}$$

Per-class precision, recall, and F1-score are derived from the confusion matrix for the three classes (low, moderate, high). For class c,

$$\begin{aligned} \text{Precision}_c &= \frac{\text{TP}_c}{\text{TP}_c + \text{FP}_c}, \\ \text{Recall}_c &= \frac{\text{TP}_c}{\text{TP}_c + \text{FN}_c}, \\ \text{F1}_c &= \frac{2\,\text{Precision}_c \cdot \text{Recall}_c}{\text{Precision}_c + \text{Recall}_c}. \end{aligned} \tag{4}$$

Macro-averaged precision, recall, and F1 are obtained by averaging the per-class values:

$$\begin{aligned} \text{Macro-Precision} &= \tfrac{1}{3} \sum_{c \in \{\text{low, moderate, high}\}} \text{Precision}_c, \\ \text{Macro-Recall} &= \tfrac{1}{3} \sum_{c \in \{\text{low, moderate, high}\}} \text{Recall}_c, \\ \text{Macro-F1} &= \tfrac{1}{3} \sum_{c \in \{\text{low, moderate, high}\}} \text{F1}_c. \end{aligned} \tag{5}$$

For each configuration (model, S), we present: (i) overall accuracy (Eq. 3), (ii) macro-averaged precision, recall, and F1 (Eq. 5), (iii) the per-class precision/recall/F1 (Eq. 4), and (iv) the 3×3 confusion matrix. This set jointly captures both aggregate performance and the behavior on individual classes,

 doi:10.20944/preprints202509.0146.v1

which is important given the class distribution reported in Section 3.3. Results are reported separately for each temporal shift S to make the effect of segmentation density directly comparable across models.

4. Results and Discussion

4.1. Effect of Temporal Shift on Aggregate Metrics (RQ1)

Figure 4 plots accuracy (left) and macro-F1 (right) as a function of temporal shift S for each of the six models (one panel per model). Across models, both accuracy and macro-F1 improve as the shift decreases from 1.00 s to 0.25 s. The trend is visible for all six classifiers and is monotonic in most cases. This pattern is consistent with the change in effective sample count introduced by S (Section 3.3): at fixed L and recording duration, reducing S increases the number of windows per recording (from 147 at S=128 to 585 at S=32), and therefore increases the number of training and test examples used by the models. Because windows created with small S share a large portion of samples, the additional examples are temporally denser rather than independent, but they still yield higher aggregate scores on the held-out test set.

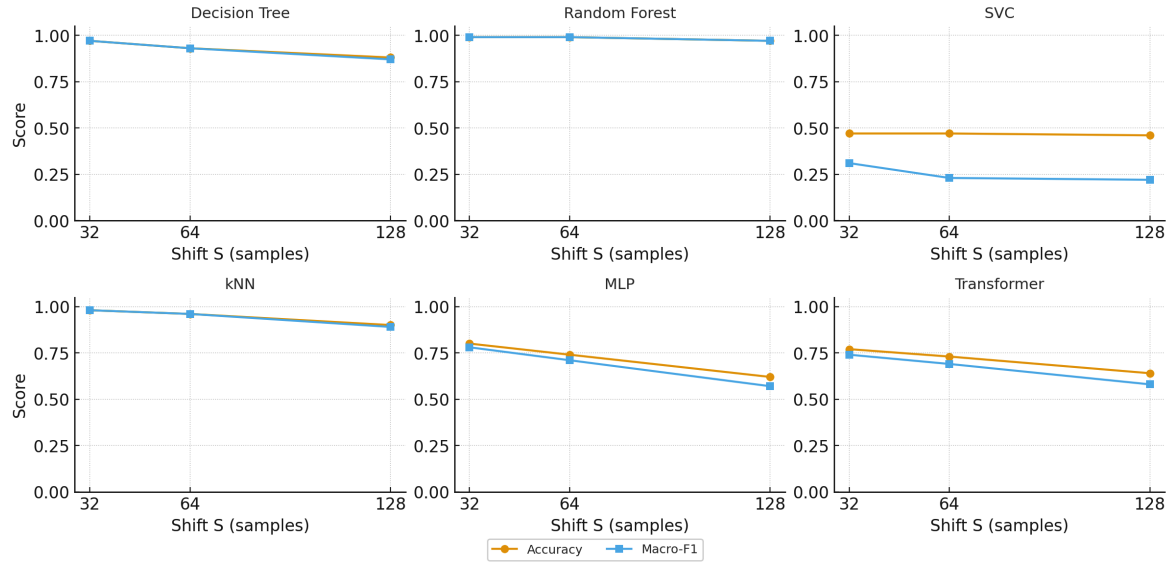

Figure 4. Accuracy and macro-F1 versus temporal shift S.

The macro-F1 curves generally follow the same direction as accuracy, indicating that the improvement with smaller S is not confined to a single class. The following Section 4.2 analyzes class-wise behavior in detail and shows how shifts in S alter the confusion structure. Here, at the aggregate level, the main observation is that using a smaller shift produces higher accuracy and macro-F1 for all evaluated classifiers under the same preprocessing and features. When comparing models or reporting baselines, it is therefore important to keep S fixed; otherwise, differences in performance may reflect segmentation density rather than properties of the classifiers.

4.2. Class-Wise Effects (RQ2)

We analyzed how temporal shift S affected each class (low, moderate, high) when the window length was fixed. Figures 5-10 show the per-class F1 results for the models across the three shifts. Across the six classifiers, the grouped bar charts show the same overall tendency: per-class F1 increases as the shift S decreases. The effect is most pronounced for Random Forest, kNN, and Decision Tree (clear gains from S=128 to 64, followed by smaller but positive steps at S=32). The MLP and the transformer also improve across all three classes, with the moderate class remaining the lowest at each S. The SVC is the outlier: its gains are modest and mostly appear at S=32, and class separation—especially for the moderate label—remains limited. These trends align with the confusion matrices, where decreasing S raises the diagonal counts and reduces moderate–neighbor errors.

 doi:10.20944/preprints202509.0146.v1

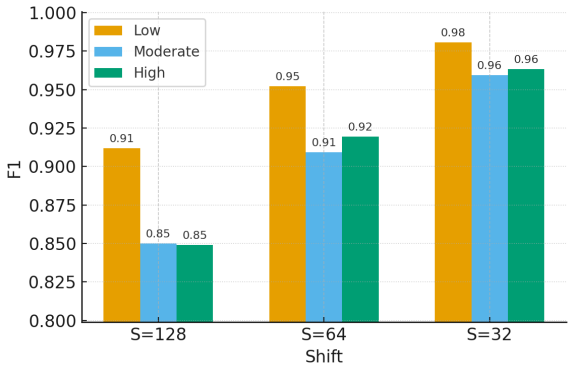

Figure 5. Per-class F1 for the Decision Tree model for different time window shifts.

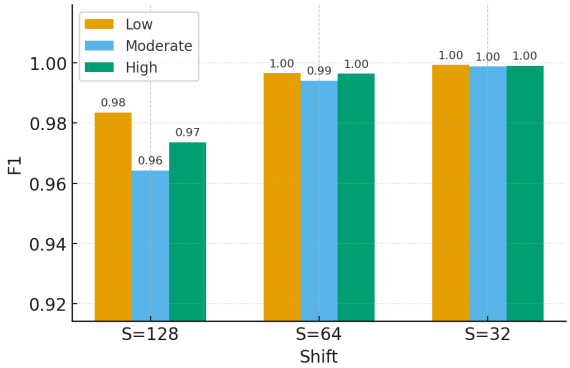

Figure 6. Per-class F1 for the Random Forest model for different time window shifts.

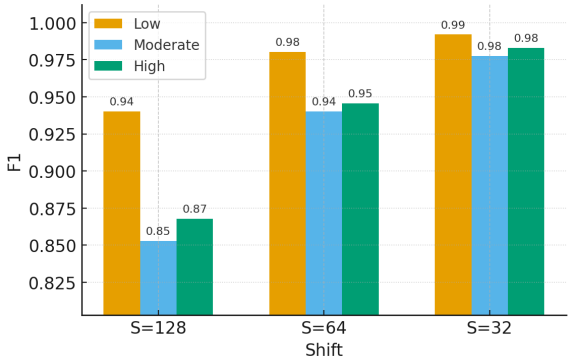

Figure 7. Per-class F1 for the SVC model for different time window shifts.

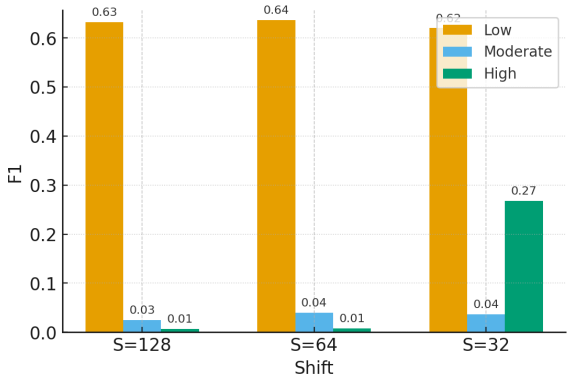

Figure 8. Per-class F1 for the kNN model for different time window shifts.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 2 September 2025 doi:10.20944/preprints202509.0146.v1

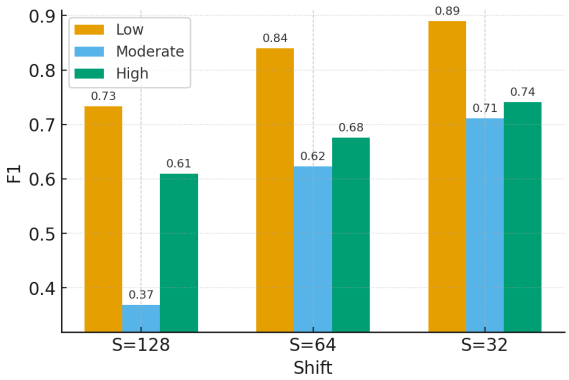

Figure 9. Per-class F1 for the MLP model for different time window shifts.

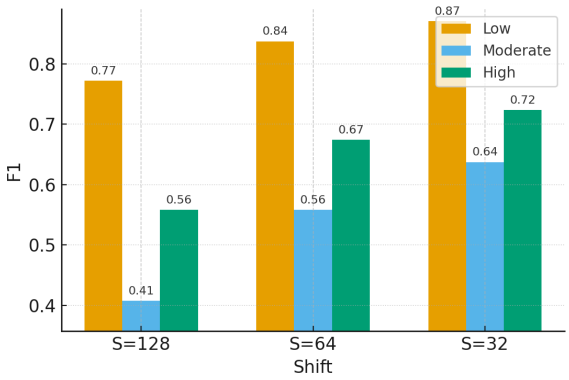

Figure 10. Per-class F1 for the transformer model for different time window shifts.

The confusion matrices (Figures 11-16) corroborate these trends. For the Random Forest (Figure 12), off-diagonal mass concentrates around confusions between *moderate* and its neighbors at S=128; as S decreases to 64 and 32, diagonal counts rise and moderate–neighbor confusions shrink. Analogous patterns appear for Decision Tree and kNN (Figures 11 and 13). The MLP and the transformer improve with smaller S, yet retain more residual moderate–neighbor confusions (Figures 15 and 16). The SVC stands out with weaker class separation overall, particularly at S=32 (Figure 14).

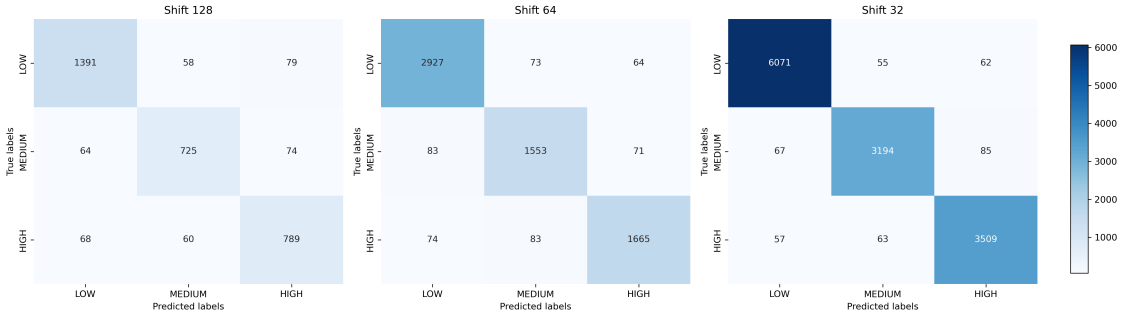

Figure 11. Confusion matrices for the DecisionTree model for different time window shifts.

 doi:10.20944/preprints202509.0146.v1

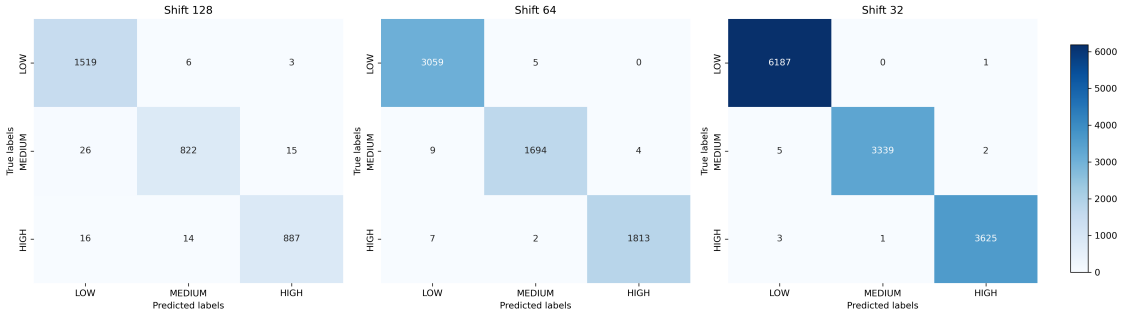

Figure 12. Confusion matrices for the RandomForest model for different time window shifts.

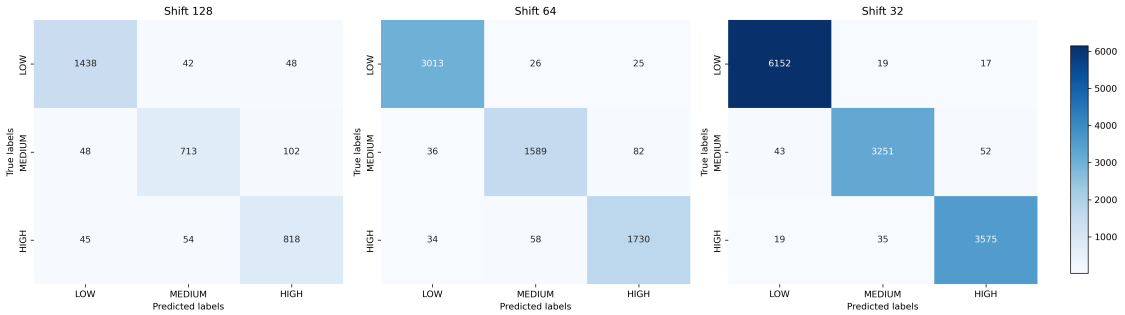

Figure 13. Confusion matrices for the kNN model for different time window shifts.

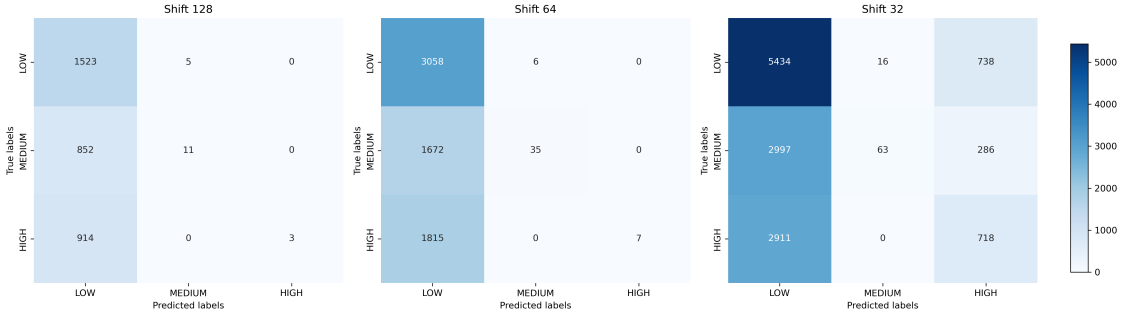

Figure 14. Confusion matrices for the SVC model for different time window shifts.

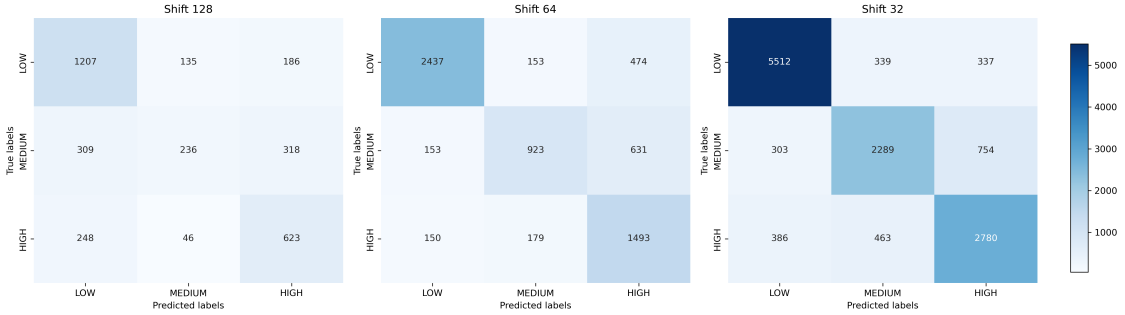

Figure 15. Confusion matrices for the MLP model for different time window shifts.

 doi:10.20944/preprints202509.0146.v1

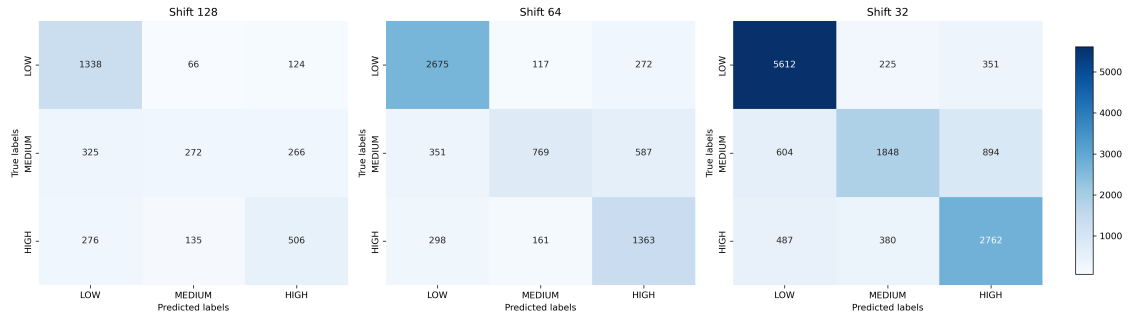

Figure 16. Confusion matrices for the transformer model for different time window shifts.

Taken together, the class-wise view makes clear that the choice of S changes not only aggregate scores but also the distribution of errors — chiefly by reducing moderate–neighbor confusions as sampling becomes denser. Reporting per-class metrics and confusion matrices alongside aggregate numbers helps attribute improvements to segmentation settings versus model properties.

4.3. Data Volume and Sample Independence (RQ3)

With fixed window length, the temporal shift S determines both the overlap between adjacent windows and how many windows are produced from the same recording (see Section 3.3). On the held-out test split, the total number of windows grows from 3308 at S=128 to 6593 at S=64 and to 13163 at S=32. Class totals scale accordingly: *low* 1528 $\rightarrow$ 3064 $\rightarrow$ 6188, *moderate* 863 $\rightarrow$ 1707 $\rightarrow$ 3346, and *high* 917 $\rightarrow$ 1822 $\rightarrow$ 3629. Thus, halving S approximately doubles the number of available windows while leaving the class mix essentially unchanged.

The aggregate gains reported in Sections 4.1–4.2 are therefore naturally linked to denser sampling: smaller S supplies more training and evaluation examples without altering labels or features. These extra examples are not independent, however. At fixed L, adjacent windows share a large fraction of samples, which increases within-subject autocorrelation at the window level. Under the subject-wise split used here, this densification does not create train–test leakage across participants, but higher scores at smaller S should be interpreted as benefits of finer temporal sampling rather than evidence that models receive fundamentally new, independent observations.

Practically, the choice of S should match the study goal and be reported alongside L. For responsive tracking or when models stabilize only with many examples, smaller S is advantageous because it increases sample count at both training and test. For conservative benchmarking or when independence of observations is paramount, larger S reduces temporal redundancy and tightens the effective sample size, at the cost of some performance. In all cases, S must be kept fixed when comparing classifiers so that differences in metrics reflect model properties rather than segmentation density.

4.4. Model Dependence (RQ4)

This section contrasts how the six classifiers respond to the temporal shift S under the same preprocessing and feature set. Table 3 compiles accuracy and macro-averaged precision, recall, and F1 for $S \in \{128, 64, 32\}$.

Across architectures, the direction of the effect is consistent: smaller S improves aggregate metrics. The magnitude, however, is model-dependent. Tree-based learners (Decision Tree, Random Forest) and kNN show the largest, nearly monotonic gains, reaching near-ceiling performance at S=32. The MLP and the transformer also benefit from denser sampling across all classes, yet remain below the tree-based baselines at each S. The SVC configuration used here exhibits the weakest improvements and limited class separation even at S=32. These differences align with the class-wise analyses in Section 4.2, where the *moderate* class remains the most challenging for all models but improves as S decreases.

 doi:10.20944/preprints202509.0146.v1

From a computational standpoint, reducing S multiplies the number of window-level examples (Section 4.3), which increases training and evaluation cost. The impact is model-specific: kNN scales prediction cost with the size of the training set; tree ensembles increase fit time with the number of samples and trees; neural models (MLP, transformer) require more updates per epoch. In practice, selecting S therefore trades off accuracy gains from denser sampling against computational budget. Comparisons between models should keep (L, S) fixed so that differences in metrics reflect model properties rather than segmentation density.

Table 3. Performance of classification models across temporal shifts (L=512 samples).

	Shift = 128				Shift = 64				Shift = 32			
Model	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
Decision Tree	0.88	0.87	0.87	0.87	0.93	0.93	0.93	0.93	0.97	0.97	0.97	0.97
Random Forest	**0.97**	**0.98**	**0.97**	**0.97**	**0.99**	**0.99**	**0.99**	**0.99**	**0.99**	**0.99**	**0.99**	**0.99**
SVC	0.46	0.72	0.34	0.22	0.47	0.77	0.34	0.23	0.47	0.56	0.36	0.31
kNN	0.90	0.89	0.89	0.89	0.96	0.96	0.95	0.96	0.98	0.98	0.98	0.98
MLP	0.62	0.60	0.58	0.57	0.74	0.73	0.72	0.71	0.80	0.78	0.78	0.78
Transformer	0.64	0.61	0.58	0.58	0.73	0.72	0.69	0.69	0.77	0.76	0.74	0.74

5. Conclusions

This study examined the role of the temporal window shift S in EEG classification of cognitive fatigue (operationalized here via mental workload ratings in the STEW dataset) under a fixed window length and a subject-wise evaluation protocol. Across six classifiers, smaller shifts—thus denser segmentation—consistently increased aggregate metrics (accuracy and macro-F1), with the strongest improvements for tree-based models and k-NN and clear, though smaller, gains for the MLP and the transformer. Class-wise analyses showed that the moderate label remained the most difficult at all S, yet its confusion with neighboring classes diminished as S decreased. These patterns indicate that segmentation density is a consequential design choice that should be controlled and reported alongside model details and the evaluation regime.

Based on these findings, we recommend that future EEG workload/fatigue studies: (i) report both window length L and shift S (and the implied overlap ratio) together with the evaluation protocol; (ii) keep (L, S) fixed when comparing models so that differences reflect model properties rather than data densification; (iii) accompany aggregate metrics with per-class scores and confusion matrices to expose how class-wise errors change with segmentation; and (iv) when applicable, state the number of repetitions/seeds and summarize variability across runs or subjects. These practices should improve reproducibility and make cross-paper comparisons more reliable.

Choosing S should follow the study goal. When responsiveness and data volume are critical—for example, to stabilize training of models that benefit from many examples—smaller S is advantageous because it increases the number of windows without altering labels. When independence of observations and computational economy are priorities, larger S reduces temporal redundancy and tightens the effective sample size, with an expected trade-off in aggregate scores. In all cases, (L, S) should be disclosed and justified.

This work has several limitations. It analyzes a single dataset, a single window length, and a fixed feature-extraction pipeline; it does not include repeated runs or statistical testing, and all results are offline. Labels derive from self-reported workload levels, which may introduce subjectivity, and we did not explore architecture-specific tuning beyond standard configurations. These constraints delimit the scope of our conclusions but do not affect the central observation that the shift parameter materially shapes reported performance. These limitations motivate several avenues for future work.

Future work will extend the analysis across datasets and tasks, probe a broader grid of (L, S) combinations (including longer and shorter windows), and incorporate repeated evaluations with statistical testing. It will also examine architectures tailored to EEG sequences (e.g., convolutional

 doi:10.20944/preprints202509.0146.v1

and recurrent variants and transformer hybrids) under the same shift-controlled protocol, quantify effective sample size under overlap, and evaluate online settings where latency and computational budget interact directly with the choice of S.

Author Contributions: Conceptualization, A.W.; methodology, A.W.; software, M.S.; validation, A.W. and M.S; formal analysis, A.W.; investigation, A.W. and K.Ż; resources, A.W., M.S. and K.Ż.; data curation, M.S.; writing—original draft preparation, A.W.; writing—review and editing, A.W. and K.Ż.; visualization, A.W. and M.S.; supervision, A.W.; project administration, A.W.. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Ethical review and approval were waived for this study due to the retrospective nature of the research and the use of anonymized datasets, which do not involve identifiable human participants.

Informed Consent Statement: Not applicable. Participant consent was obtained by the STEW dataset authors; this study used the anonymized public release.

Data Availability Statement: The data presented in this study are openly available online at https://ieee-dataport.org/open-access/stew-simultaneous-task-eeg-workload-dataset (accessed on 20 July 2025).

Acknowledgments: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Lim, W.L.; Sourina, O.; Wang, L.P. STEW: Simultaneous Task EEG Workload Data Set. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* **2018**, *26*, 2106–2114. https://doi.org/10.1109/TNSRE.2018.2872924.
2. Lim, W.L.; Sourina, O.; Wang, L. STEW: Simultaneous Task EEG Workload Dataset, 2018. https://doi.org/10.21227/44r8-ya50.
3. Brookshire, G.; Kasper, J.; Blauch, N.M.; Wu, Y.C.; Glatt, R.; Merrill, D.A.; Gerrol, S.; Yoder, K.J.; Quirk, C.; Lucero, C. Data leakage in deep learning studies of translational EEG. *Frontiers in Neuroscience* **2024**, *18*, 1373515. https://doi.org/10.3389/fnins.2024.1373515.
4. Lee, H.T.; Cheon, H.R.; Lee, S.H.; Shim, M.; Hwang, H.J. Risk of data leakage in estimating the diagnostic performance of a deep-learning-based CAD system for psychiatric disorders. *Scientific Reports* **2023**, *13*. https://doi.org/10.1038/s41598-023-43542-8.
5. Del Pup, F.; Zanola, A.; Tshimanga, L.F.; Bertoldo, A.; Finos, L.; Atzori, M. The role of data partitioning on the performance of EEG-based deep learning models in supervised cross-subject analysis: A preliminary study. *Computer Methods and Programs in Biomedicine* **2025**, *246*, 107808. https://doi.org/10.1016/j.cmpb.2025.107808.
6. Falih, M.H.; Alshamasin, M.S.; Abukhurma, R.; Khdour, T. Impact of Sliding Window Overlap Ratio on EEG-Based ASD Diagnosis Using Machine Learning Techniques. *Applied Sciences* **2024**, *14*, 11702. https://doi.org/10.3390/app142411702.
7. Alghanim, M.; Abd El-Latif, A.A.; et al. A Hybrid Deep Neural Network Approach to Recognize Brain Fatigue. *Computational Intelligence and Neuroscience* **2024**. https://doi.org/10.1155/2024/9898333.
8. Christou, V.; Miltiadous, A.; Tsoulos, I.; Karvounis, E.; Tzimourta, K.D.; Tsipouras, M.G.; Anastasopoulos, N.; Tzallas, A.T.; Giannakeas, N. Evaluating the Window Size's Role in Automatic EEG Epilepsy Detection. *Sensors* **2022**, *22*, 9233. https://doi.org/10.3390/s22239233.
9. Yuvaraj, R.; Samyuktha, S.; Fogarty, J.; Huang, J.S.; Tan, S.; Wong, T.K. Optimal EEG Time Window Length for Boredom Classification using Combined Non-linear Features. In Proceedings of the Proceedings of EUSIPCO 2024, Lyon, France, 2024; pp. 1756–1761. ISBN: 978-9-4645-9361-7.
10. Safari, M.; Shalbaf, R.; Bagherzadeh, S.; Shalbaf, A. Classification of mental workload using brain connectivity and machine learning on electroencephalogram data. *Scientific Reports* **2024**, *14*, 9153. https://doi.org/10.1038/s41598-024-59652-w.
11. Sun, C.; Mou, C. Survey on the research direction of EEG-based signal processing. *Frontiers in Neuroscience* **2023**, *17*, 1203059. https://doi.org/10.3389/fnins.2023.1203059.

 doi:10.20944/preprints202509.0146.v1

12. Hamzah, H.A.; Atia, A.; Serag, A.; Elmisery, A.M.; Ware, A.; Khan, S.; Ma, J. EEG-based emotion recognition systems: Comprehensive review of feature extraction methods. *Heliyon* **2024**, *10*, e31686. https://doi.org/10.1016/j.heliyon.2024.e31686.
13. Demirezen, G.; Temizel, T.T.; Brouwer, A.M. Reproducible machine learning research in mental workload classification using EEG. *Frontiers in Neuroergonomics* **2024**, *5*, 1346794. https://doi.org/10.3389/fnrgo.2024.1346794.
14. Imran, M.A.A.; Nasirzadeh, F.; Karmakar, C. Designing a practical fatigue detection system: A review on recent developments and challenges. *Journal of Safety Research* **2024**, *90*, 100–114. https://doi.org/10.1016/j.jsr.2024.05.015.
15. Afzal, M.A.; Gu, Z.; Bukhari, S.U.; Afzal, B. Brainwaves in the Cloud: Cognitive Workload Monitoring Using Deep Gated Neural Network and Industrial Internet of Things. *Applied Sciences* **2024**, *14*, 5830. https://doi.org/10.3390/app14135830.
16. Vafaei, E.; Lee, J. Transformers in EEG Analysis: A Review of Architectures and Applications in Motor Imagery, Seizure, and Emotion Classification. *Sensors* **2025**, *25*, 1293. https://doi.org/10.3390/s25051293.
17. Pfeffer, M.A.; Ling, S.S.H.; Wong, J.K.W. Exploring the frontier: Transformer-based models in EEG signal analysis for brain–computer interfaces. *Computers in Biology and Medicine* **2024**, *178*, 108705. https://doi.org/10.1016/j.cmpb.2024.108705.
18. Zeynali, M.; Seyedarabi, H.; Afrouzian, R. Classification of EEG signals using Transformer-based deep learning and ensemble models. *Biomedical Signal Processing and Control* **2023**, *86*, 105130. https://doi.org/10.1016/j.bspc.2023.105130.
19. Yao, X.; Li, T.; Ding, P.; Wang, F.; Zhao, L.; Gong, A.; Nan, W.; Fu, Y. Emotion Classification Based on Transformer and CNN for EEG Spatial–Temporal Feature Learning. *Brain Sciences* **2024**, *14*, 268. https://doi.org/10.3390/brainsci14030268.
20. Si, X.; Huang, D.; Sun, Y.; Ming, D. Temporal Aware Mixed Attention-based Convolution and Transformer Network (MACTN) for EEG Emotion Recognition. *Computers in Biology and Medicine* **2024**, *171*, 108091. https://doi.org/10.1016/j.cmpb.2024.108091.
21. Cheng, Z.; Du, W.; Li, Y.; Zhang, Z.; Zheng, W. EEG-based emotion recognition using multi-scale dynamic 1D CNN and gated Transformer. *Scientific Reports* **2024**, *14*, 18011. https://doi.org/10.1038/s41598-024-82705-z.
22. Xu, Y.; Sha, Y.; Chen, F.; Chen, X.; Wu, X.; Xu, P. AMDET: Attention-Based Multiple Dimensions EEG Transformer. *IEEE Transactions on Affective Computing* **2024**, *15*, 2293–2307. https://doi.org/10.1109/TAFFC.2023.3321576.
23. Liang, S.; Li, L.; Zu, W.; Feng, W.; Hang, W. Adaptive deep feature representation learning for cross-subject EEG decoding. *BMC Bioinformatics* **2024**, *25*, 393. https://doi.org/10.1186/s12859-024-06024-w.
24. Joshi, A.; Kale, S.; Chandel, S.; Pal, D.K. Likert Scale: Explored and Explained. *British Journal of Applied Science & Technology* **2015**, *7*, 396–403. https://doi.org/10.9734/BJAST/2015/14975.
25. de Cheveigné, A.; Nelken, I. Filters: When, Why, and How (Not) to Use Them. *Neuron* **2019**, *102*, 280–293. https://doi.org/10.1016/j.neuron.2019.02.039.
26. Singh, A.K.; Krishnan, S. Trends in EEG signal feature extraction applications. *Frontiers in Artificial Intelligence* **2023**, *5*, 1072801. https://doi.org/10.3389/frai.2022.1072801.
27. Ranjan, R.; Chandra Sahana, B.; Kumar Bhandari, A. Ocular artifact elimination from electroencephalography signals: A systematic review. *Biocybernetics and Biomedical Engineering* **2021**, *41*, 960–996. https://doi.org/10.1016/j.bbe.2021.06.007.
28. Frølich, L.; Dowding, I. Removal of muscular artifacts in EEG signals: a comparison of linear decomposition methods. *Brain informatics* **2018**, *5*, 13–22. https://doi.org/10.1007/s40708-017-0074-6.
29. Kang, T.; Thielen, J.; Vidaurre, C.; Lemm, S. ICA-based EEG artifact removal does not improve deep decoding performance. *Journal of Neural Engineering* **2024**, *21*, 056015. https://doi.org/10.1088/1741-2552/ad6a13.
30. Mutanen, T.P.; Mäkinen, J.; Ilmoniemi, R.; Rocchi, L.; Nieminen, J.O. A simulation study: comparing independent component analysis and signal-space projection–source-informed reconstruction for rejecting muscle artifacts evoked by TMS. *Frontiers in Human Neuroscience* **2024**, *18*, 1324958. https://doi.org/10.3389/fnhum.2024.1324958.
31. Guo, T.; Zhang, T.; Lim, E.; López-Benítez, M.; Ma, F.; Yu, L. A Review of Wavelet Analysis and Its Applications: Challenges and Opportunities. *IEEE Access* **2022**, *10*, 58869–58903. https://doi.org/10.1109/ACCESS.2022.3179517.
32. Acharya, S.; Patel, N.; Subramanian, R. A systematic review of EEG-based automated mental stress quantification. *Applied Soft Computing* **2025**, *159*, 111573. https://doi.org/10.1016/j.asoc.2025.111573.

 doi:10.20944/preprints202509.0146.v1

33. Nayak, A.B.; Shah, A.; Maheshwari, S.; Anand, V.; Chakraborty, S.; Kumar, T.S. An empirical wavelet transform-based approach for motion artifact removal in electroencephalogram signals. *Decision Analytics Journal* **2024**, *10*, 100420. https://doi.org/10.1016/j.dajour.2024.100420.
34. Mehmood, I.; Li, H.; Umer, W.; Arsalan, A.; Anwer, S.; Mirza, M.A.; Ma, J.; Antwi-Afari, M.F. Multimodal integration for data-driven classification of mental fatigue during construction equipment operations: Incorporating electroencephalography, electrodermal activity, and video signals. *Developments in the Built Environment* **2023**, *15*, 100198. https://doi.org/10.1016/j.dibe.2023.100198.
35. Campos, M.S.R.; McCracken, H.S.; Uribe-Quevedo, A.; Grant, B.L.; Yielder, P.C.; Murphy, B.A. A Machine Learning Approach to Classifying EEG Data Collected with or without Haptic Feedback during a Simulated Drilling Task. *Brain Sciences* **2024**, *14*, 894. https://doi.org/10.3390/brainsci14090894.
36. Guo, H.; Chen, S.; Zhou, Y.; Xu, T.; Zhang, Y.; Ding, H. A hybrid critical channels and optimal feature subset selection framework for EEG fatigue recognition. *Scientific Reports* **2025**, *15*, 2139. https://doi.org/10.1038/s41598-025-86234-1.
37. Avola, D.; Cascio, M.; Cinque, L.; Fagioli, A.; Foresti, G.L.; Marini, M.R.; Pannone, D. Analyzing EEG data with machine and deep learning: A benchmark. In Proceedings of the International conference on image analysis and processing. Springer, 2022, pp. 335–345. https://doi.org/10.48550/arXiv.2203.10009.
38. Vafaei, E.; Hosseini, M. Transformers in EEG Analysis: A Review of Architectures and Applications in Motor Imagery, Seizure, and Emotion Classification. *Sensors* **2025**, *25*. https://doi.org/10.3390/s25051293.
39. Siddhad, G.; Gupta, A.; Dogra, D.P.; Roy, P.P. Efficacy of Transformer Networks for Classification of Raw EEG Data. *Biomedical Signal Processing and Control* **2024**, *85*, 105488.