**Synthesis: A Multidisciplinary Research Journal**
Volume 2, Issue 1, January - March 2024, Pp: 44-54
© 2024, All Rights Reserved @ Macaw Publications

***Research Paper***

# Real-Time Cognitive Load Estimation in Augmented Reality Interfaces Using EEG-Driven Adaptive Algorithms

[1*] Sreekanth Rallapalli, [2] Murtuza Ahamed Khan

*[1*] Professor and HoD, Department of Master of Computer Applications, Nitte Meenakshi Institute of Technology, Bengaluru, Karnataka, India*
*Email: Sreekanth.rallapalli@nmit.ac.in*
*[2] Lecturer, Department of Computer Engineering, College of Computer Science, King Khalid University, Abha, Saudi Arabia*
*Email: murtuza@kku.edu.sa*

**Corresponding Author(s): Sreekanth.rallapalli@nmit.ac.in*

**Article Info**
Received: 19/10/2023
Revised: 05/02/2024
Accepted:17/03/2024
Published:31/03/2024

## Abstract

Augmented reality (AR) systems increasingly support complex decision-making and user interaction, but high visual density often results in cognitive overload, reducing performance and increasing fatigue. Traditional methods for estimating cognitive load are either retrospective or too invasive for seamless real-time adaptation. This study aims to develop a real-time, EEG-driven cognitive load estimation model integrated with an adaptive AR interface to dynamically adjust content based on user mental effort. A hybrid CNN–LSTM deep learning architecture was trained on the STEW EEG dataset using band-specific power spectral density and Hjorth features extracted from preprocessed signals. The model was evaluated using subject-wise 5-fold cross-validation and deployed in an AR environment using TensorFlow Lite on Microsoft HoloLens 2. Real-time inference was achieved with a latency of 24.5 ms and a deployable model size of 9.2 MB. The proposed model achieved 90% classification accuracy and a macro-averaged F1-score of 89%, outperforming SVM, Random Forest, and CNN baselines. Usability tests in AR showed a 31.7% reduction in NASA-TLX scores, a 25.4% decrease in task error rates, and a 14.8-second improvement in task completion time using multi-class adaptive UI. This research demonstrates that real-time EEG-based cognitive estimation can significantly enhance user experience in AR systems. The proposed framework offers a scalable and efficient solution for neuroadaptive interfaces in training, healthcare, and industrial applications.

**Keywords:** Cognitive Load Estimation, EEG, Augmented Reality, CNN–LSTM, Adaptive Interface, Brain–Computer Interface, Real-Time Classification, Usability Evaluation.

## 1. Introduction

Augmented Reality (AR) technologies have seen rapid integration across a range of industries, including healthcare, manufacturing, education, and defense. By overlaying digital content onto the real-world environment, AR systems provide enhanced interactive experiences that support task execution, decision-making, and immersive learning. Despite the transformative potential of AR interfaces, their usability and effectiveness are increasingly challenged by cognitive overload experienced by users. As the complexity and density of visual information within AR environments grow, users often face difficulties in processing and responding to these stimuli efficiently. This leads to increased task completion time, errors in interaction, and overall user fatigue, particularly in mission-critical or high-pressure scenarios. Consequently, there is a growing need to embed real-time cognitive load estimation into AR interfaces, allowing systems to adapt dynamically to users' mental states.

Cognitive load represents the mental effort required to process information in a given context. In AR environments, this load is influenced by a multitude of factors including interface design, information density, user experience level, and task complexity. The ability to accurately estimate cognitive load in real time has the potential to revolutionize user-centered interface design by enabling adaptive systems that tailor content delivery, interaction modalities, and visual feedback based on users' real-time mental state. However, traditional approaches for assessing cognitive load—such as NASA-TLX questionnaires, eye-tracking, and behavioral observation—are inherently retrospective, subjective, and unsuitable for seamless integration in real-time AR systems.

Recent advancements in Brain-Computer Interfaces (BCIs), particularly the use of electroencephalogram (EEG) signals, provide a non-invasive and continuous method to assess mental workload. EEG signals reflect neural activity and have been shown to correlate strongly with cognitive states such as attention, engagement, and workload. The frequency-domain characteristics of EEG—such as increased theta activity and suppressed alpha waves—are directly linked to variations in cognitive effort. Several research initiatives have leveraged EEG to monitor cognitive load in controlled lab environments; however, few have succeeded in translating this into robust, real-time, and scalable solutions for real-world AR interfaces.

One of the main challenges in EEG-driven cognitive load estimation is the dynamic and non-stationary nature of brain signals, which vary significantly across individuals, tasks, and contexts. Moreover, the integration of such systems into AR environments presents additional hurdles including signal noise, latency, hardware compatibility, and user comfort. Many existing solutions rely on static machine learning models that lack generalization, require large labeled datasets, and fail to adapt to evolving user states. These limitations necessitate a shift toward real-time adaptive algorithms capable of learning from continuous EEG streams, adjusting prediction boundaries, and personalizing inference over time.

Emerging research in federated learning and distributed edge computing has contributed significantly to privacy-preserving, real-time analytics, especially in constrained and decentralized environments. For example, multiple studies have demonstrated the effectiveness of federated architectures in intrusion detection systems (IDS) within Internet of Things (IoT) ecosystems, where data remains on local devices while models are updated collaboratively [1]–[3]. This paradigm has inspired a shift toward decentralized learning strategies for sensitive biomedical applications, including mental state detection via EEG. In such setups, the user's cognitive patterns are analyzed without exposing raw data, enabling both privacy and adaptability.

Additionally, hybrid models incorporating ensemble learning, multi-view fusion, and federated deep learning have shown significant performance improvements in highly dynamic and heterogeneous environments. These studies highlight the need for lightweight yet intelligent agents that operate with minimal delay and offer contextual intelligence. Although primarily applied in cybersecurity and industrial automation, these methodologies offer relevant insights for developing adaptive AR interfaces driven by neurophysiological data. For instance, ensemble models combining spatial and temporal features have demonstrated high detection accuracy in noisy environments—an important parallel for EEG-based cognitive load estimation where real-world signal contamination is a persistent issue [4], [5].

Moreover, privacy-preserving techniques such as differential privacy, secure aggregation, and homomorphic encryption have been adopted to enhance data confidentiality in decentralized AI frameworks [6], [7], [8]. When integrated into EEG-based systems, these mechanisms allow secure model updates without compromising individual neuro-data. This is crucial for AR applications in healthcare or education, where user consent and ethical data handling are non-negotiable. Recent advancements have also seen the adoption of blockchain-based logging for auditability and trust in federated training pipelines [9], [10].

The fusion of these technological domains—EEG-based neuroanalysis, real-time adaptive algorithms, and AR interface engineering—creates a fertile ground for research and innovation. While prior work in federated learning for IoT intrusion detection has focused on anomaly patterns in network traffic, this study applies similar distributed learning mechanisms to human EEG patterns, thus repurposing established models into the context of cognitive state estimation. A comprehensive review of centralized versus federated learning in IoT environments shows that on-device intelligence leads to superior latency performance and improved system resilience [11]. This insight supports the architectural decision to process EEG data at the edge, close to the AR device, minimizing reliance on cloud infrastructure and enabling responsive user feedback.

Despite promising progress, several technical gaps remain unaddressed in the literature. First, there is limited research on integrating EEG signal analysis with AR interface elements such as 3D object placement, holographic cues, or voice-overlays in real time. Second, the personalization of cognitive thresholds—adapting the model to an individual user's baseline stress or attention level—is often overlooked. Lastly, the majority of existing models are optimized offline, lacking the ability to evolve with continued user interaction or context-switching. This creates an opportunity to propose a novel EEG-driven cognitive load estimation framework that not only detects user states in real time but also feeds this intelligence back into the AR interface to adjust content complexity, visual density, and interaction prompts.

In this study, we propose a real-time, EEG-driven cognitive load estimation framework for AR environments, designed using adaptive machine learning algorithms and privacy-preserving signal processing techniques. The framework dynamically learns from user EEG data streams, identifies cognitive state transitions, and triggers AR interface adjustments to maintain optimal cognitive balance. Through rigorous experimentation involving interactive AR tasks and EEG monitoring, we evaluate model performance, interface responsiveness, and overall usability improvements. Our system is lightweight, modular, and capable of operating on edge devices with minimal computational overhead.

The key contributions of this study are as follows:

- *Real-Time Cognitive Load Estimation:* We develop and validate an adaptive algorithm that processes EEG signals in real time to identify cognitive load levels, achieving high temporal precision without requiring labeled calibration data per user.
- *EEG-Driven AR Adaptation:* The system dynamically adjusts AR interface components—such as object density, text size, and task prompts—based on real-time cognitive feedback, enhancing user engagement and reducing overload.
- *Privacy-Preserving Architecture:* Leveraging principles from federated learning and differential privacy, our solution ensures that sensitive EEG data remains decentralized and securely processed at the edge.

In the following sections, we outline the related work in EEG-based cognitive modeling and adaptive AR interfaces (Section II), describe the technical methodology including signal acquisition and model design (Section III), present experimental setups and evaluation metrics (Section IV), discuss results with a focus on usability and system performance (Section V), and conclude with key findings and future research directions (Section VI).

## 2. Related Work

The integration of real-time cognitive state estimation into augmented reality (AR) interfaces necessitates cross-domain research involving human-computer interaction, EEG signal processing, and adaptive learning frameworks. Although direct applications of EEG-based cognitive load estimation in AR remain sparse, valuable insights can be derived from closely aligned domains such as intrusion detection systems (IDS), Industrial Internet of Things (IIoT), and federated learning frameworks. These fields have progressively incorporated privacy-preserving intelligence and adaptive learning mechanisms, both of which are critical for EEG-based AR systems where user data is highly sensitive and signal dynamics evolve rapidly.

### *2.1 Federated Learning in Intrusion Detection: Architecture and Strengths*

Federated learning (FL) has emerged as a decentralized machine learning paradigm particularly suited for environments where data privacy, low latency, and distributed intelligence are essential. Several works have explored FL-based intrusion detection in IIoT systems, emphasizing its robustness against data leakage and its capability to aggregate knowledge across multiple local models without transmitting raw data.

Rashid et al. proposed a federated learning-based intrusion detection framework that combined local gradient updates from edge devices with a centralized aggregation server to detect network anomalies **[12]**. Their method achieved high accuracy while significantly reducing communication overhead. However, the system was designed primarily for packet-based data and did not incorporate continuous-time signals like EEG. Similarly, Pradeep et al. addressed optimization in task offloading for IoT environments and highlighted the importance of edge-local learning to enhance latency-sensitive systems **[13]**, a consideration directly relevant to EEG-driven AR systems that must respond instantaneously to user mental states.

Ruzafa-Alcázar et al. applied FL to industrial IoT security, focusing on privacy-preserving training and efficient model synchronization **[14]**. Although the model performed well in constrained network environments, it lacked adaptability to non-stationary signal behaviors, a major limitation when applied to dynamic EEG inputs. Agrawal et al. provided a broader review of FL architectures for IDS, stressing challenges like non-IID data, adversarial updates, and resource-constrained edge nodes **[15]**. These insights underscore the importance of lightweight, noise-tolerant learning strategies for EEG signal processing in wearable AR devices.

### *2.2 Transfer Learning and Ensemble Strategies for Real-Time Adaptation*

Recent research has attempted to address model generalization and adaptability through transfer learning and ensemble-based FL, both of which are promising for EEG-driven systems. Zhang et al. introduced a transfer learning-based federated framework to support IIoT environments with heterogeneous devices and uneven data distributions **[16]**. Their use of domain adaptation techniques aligns closely with the need to personalize EEG-based cognitive estimators for individual users in AR systems.

Tahir et al. proposed an experience-driven attack simulation system using federated anomaly detection to model evolving threat behaviors in Industry 4.0 environments **[17]**. This approach of dynamically adapting model responses based on historical patterns can inform strategies for adjusting AR interfaces in response to recurring user cognitive states. Tabassum et al. went further by employing generative adversarial networks (GANs) in conjunction with federated learning to generate synthetic attack patterns and enhance model resilience **[18]**. A similar adversarial learning framework can be applied to augment EEG datasets, addressing the scarcity of labeled training data in neuroadaptive interfaces.

Truong and Le introduced MetaCIDS, a collaborative intrusion detection framework for metaverse applications that integrates blockchain with online FL **[19]**. Their hybrid solution ensures both model transparency and low-latency updates—critical factors for cognitive AR applications where users require immediate interface feedback. Although their application domain differs, the architectural foundation of secure and real-time federated adaptation directly informs the design principles required for neuroadaptive AR systems.

Finally, Al-Marri et al. developed a federated mimic learning system to ensure privacy-preserving IDS performance without sacrificing accuracy **[20]**. Their mimic-based strategy used limited labeled samples, which is analogous to our scenario where user-specific EEG data is limited. While the system effectively learned from behavioral trends, it lacked consideration of temporal correlations—a key component in cognitive workload analysis from EEG.

### *2.3 Research Gaps and Limitations*

Despite the significant advancements in federated and adaptive learning frameworks within the context of IIoT and IDS, several critical gaps remain when translating these methodologies to cognitive load estimation in AR interfaces:

- *Lack of temporal signal modeling:* Most FL-based IDS systems are designed for static data such as packet features or access logs. EEG signals, in contrast, are highly temporal and require models that can capture spatiotemporal dependencies over sliding windows—a functionality lacking in most of the reviewed works.
- *Limited personalization and adaptation:* Cognitive load varies not only across tasks but also across users. Most IDS models apply a single global model or ensemble across nodes without personalizing thresholds or tuning parameters for each user. Such generalization is unsuitable for EEG-driven systems where individual baseline signals differ significantly.
- *Missing integration with feedback interfaces*: None of the reviewed works consider feedback-driven UI adaptation, which is essential for AR systems. FL-based models must inform not just the detection of cognitive load but also how interface parameters (e.g., visual density, text complexity) are adapted in real time.
- *Neglected cross-modal fusion:* EEG-driven AR systems can benefit from combining neurodata with contextual cues such as head movements or visual fixations. Existing FL-IDS frameworks generally rely on unimodal data sources, lacking fusion mechanisms for real-time cognitive context awareness.

### *2.4 Summary of Comparative Analysis*

The following table summarizes key characteristics of the reviewed works in terms of accuracy, adaptability, and relevance to EEG-driven cognitive load estimation.

Table 1: Comparative Analysis

| Ref | Framework & Method | Accuracy / Effectiveness | Strengths | Limitations for EEG-AR Use |
|---|---|---|---|---|
| [12] | FedAvg with local IDS | High (93%+) | Low latency, privacy-preserving | Not suited for time-series EEG |
| [13] | Edge offloading optimization | Task latency improved | Adaptive to local constraints | No cognitive state modeling |
| [14] | Privacy-enhanced federated IDS | Moderate | FL with privacy focus | Static feature modeling |
| [15] | Survey of FL for IDS | – | Detailed taxonomy | Lacks practical EEG use-case examples |
| [16] | Transfer FL with domain adaptation | High for IIoT scenarios | Heterogeneous data handling | Complex for real-time applications |
| [17] | Experience-driven anomaly design | High accuracy, dynamic | Contextual model adjustment | Not EEG-compliant or real-time ready |
| [18] | FedGAN for anomaly synthesis | Improved generalization | Augments scarce data | GANs need extensive tuning |
| [19] | Blockchain + online FL (MetaCIDS) | High, scalable | Secure, transparent | Metaverse-specific; limited EEG overlap |
| [20] | Federated mimic learning | Data-efficient | Few-shot adaptation | Weak temporal sequence handling |

### *2.5 Positioning of the Present Study:*

In contrast to the above studies, the proposed research focuses explicitly on the real-time estimation of cognitive load from EEG signals, integrated into interactive AR environments. While borrowing privacy-preserving and adaptive learning strategies from federated IDS architectures, our study innovates by addressing:

1. The real-time signal processing needs of EEG data,

2. The personalization of cognitive thresholds using online learning techniques,
3. And the interface-level adaptation within AR systems for cognitive load balancing.

This cross-disciplinary fusion of neuroadaptive interfaces, real-time federated modeling, and user-centered AR design distinguishes our contribution and fills a crucial gap in the current literature.

## 3. Methodology

This section outlines the detailed methodology adopted for designing a real-time cognitive load estimation system integrated into augmented reality (AR) interfaces using electroencephalogram (EEG) signals and adaptive learning algorithms. The methodology comprises six core components: dataset selection and preprocessing, feature extraction, model design, training and optimization, real-time AR integration, and evaluation.

### *3.1 EEG Dataset and Preprocessing*

The study utilizes the publicly available STEW dataset [21], which consists of multi-channel EEG recordings acquired during cognitive workload tasks, including arithmetic, memory recall, and Stroop tests. The dataset contains annotated labels for three workload levels: low, medium, and high, enabling supervised multi-class classification. EEG recordings were obtained using consumer-grade wearable devices at a sampling rate of 256 Hz, across multiple subjects (n ≈ 30).

The dataset exhibits a mild class imbalance (approximately 45% low, 35% medium, and 20% high workload). To address this, the Synthetic Minority Over-sampling Technique (SMOTE) was employed to generate synthetic samples of underrepresented classes. Prior to model training, EEG signals were preprocessed using a 4th-order Butterworth band-pass filter (1–40 Hz) to remove baseline drift and high-frequency noise.

Additionally, Independent Component Analysis (ICA) was applied to isolate and eliminate artifacts from eye blinks and muscle movements. The EEG signal $x_i(t)$ for channel $i$ is filtered as:

$$\tilde{x}_i(t) = \mathcal{F}_{BP}[x_i(t)], \; i = 1, 2, \ldots, C \tag{1}$$

where $\mathcal{F}_{BP}$ is the band-pass filtering operation and $C$ is the number of EEG channels.

To enable subject-independent learning, all signals were z-score normalized per subject to reduce inter-subject variability.

### *3.2 Feature Extraction*

Each preprocessed EEG signal was divided into overlapping windows of 2 seconds, with a 50% overlap. For each segment, a set of time-domain and frequency-domain features was extracted to capture signal dynamics related to cognitive effort.

#### *3.2.1 Frequency-Domain Features*

Using Welch's method, the Power Spectral Density (PSD) was calculated across four frequency bands:

- **Theta** (4–7 Hz)
- **Alpha** (8–12 Hz)
- **Beta** (13–30 Hz)
- **Gamma** (31–40 Hz)

The average power in band $b$ was computed as:

$$P_b = \frac{1}{N_b} \sum_{f \in b} |X(f)|^2 \tag{2}$$

Where $X(f)$ is the discrete Fourier transform of the EEG segment and $N_b$ is the number of frequency bins in the band $b$.

#### *3.2.2 Time-Domain Features*

Time-domain features included:

- Mean, Variance, Skewness, Kurtosis
- Hjorth Parameters: Activity, Mobility, and Complexity

For example, Hjorth Mobility is given by:

$$H_M = \sqrt{\frac{\mathrm{Var}(x'(t))}{\mathrm{Var}(x(t))}} \tag{3}$$

These features yielded a 24-dimensional feature vector per window, per channel, concatenated into a matrix for model input.

**Algorithm:** EEG Cognitive Load Estimation and AR Adaptation

**Algorithm: Real-Time EEG Cognitive Load Estimation and AR Interface Adaptation**

```
Input:
    x(t)        → Continuous EEG signal stream
    W           → Window length (e.g., 2 seconds)
    O           → Overlap ratio (e.g., 50%)
    θ           → Cognitive load classification threshold
    T_s         → Temporal smoothing window (e.g., 3 predictions)

Output:
    ŷ_t         → Estimated cognitive load label at time t
    AR_UI_state → Adjusted AR interface state

Initialize:
    Buffer ← empty list for smoothed predictions
    AR_UI_state ← default configuration

for each incoming EEG window x_i(t) with step size (W × (1 - O)) do
    1. Signal Preprocessing:
        a. Apply band-pass filter (1–40 Hz)
        b. Perform ICA for artifact removal
        c. Normalize: x̂_i(t) ← Z-score(x_i(t))

    2. Feature Extraction:
        a. Compute Power Spectral Density (PSD) in θ, α, β, γ bands
        b. Calculate Hjorth parameters (Activity, Mobility, Complexity)
        c. Extract statistical features (mean, variance,
```

```
skewness, kurtosis)
        d. Form feature vector: f_i = [P_b, H_M, μ, σ², ...]

    3. Load Classification:
        a. Feed f_i into CNN–LSTM model → output
probability vector p
        b. Determine class label: ŷ_i = argmax(p)

    4. Temporal Smoothing:
        a. Append ŷ_i to Buffer
        b. If length(Buffer) ≥ T_s:
            i. Compute mode of last T_s entries → ŷ_smooth
            ii. Remove oldest entry from Buffer

    5. AR Interface Adaptation:
        a. If ŷ_smooth = "High Load" and sustained over T_s:
            i. Reduce visual clutter, enlarge text, slow
animations
        b. Else if ŷ_smooth = "Low/Medium Load":
            i. Maintain or enhance content complexity
        c. Update AR_UI_state accordingly

end for

Return final label ŷ_t and AR_UI_state
```

Algorithm 1 outlines the step-by-step process for real-time EEG-based cognitive load estimation and dynamic AR interface adaptation. The incoming EEG signal is continuously segmented into overlapping windows and subjected to preprocessing operations such as band-pass filtering, artifact removal using ICA, and normalization. For each segment, a comprehensive feature vector is extracted by computing power spectral densities across EEG bands, Hjorth parameters, and statistical metrics. These features are then fed into a trained CNN–LSTM model, which outputs a class probability vector corresponding to cognitive load levels. To reduce prediction volatility, a temporal smoothing buffer maintains a history of recent predictions and computes the mode over a defined interval. Based on the smoothed output, the AR interface adapts in real time—simplifying the UI under high load or maintaining full complexity under lower load conditions. This pipeline ensures continuous, personalized cognitive state monitoring with immediate feedback, enhancing both user performance and system responsiveness.

### *3.3 Deep Learning Model Architecture*

A hybrid CNN–LSTM architecture was adopted to effectively capture both spatial patterns (between channels) and temporal dynamics (within signal windows).

#### *3.3.1 Convolutional Layers*

Initial layers included 1D convolutional filters for extracting spatial dependencies:

- Conv1D (filters=32, kernel=3, ReLU)
- MaxPooling1D (pool_size=2)
- BatchNormalization

#### *3.3.2 Recurrent Layers*

To model temporal dynamics, **LSTM layers** were employed:

- LSTM (64 units)
- Dropout (0.3)

#### *3.3.3 Fully Connected Layers*

The flattened feature vector passed through dense layers:

- Dense(32 units, ReLU)
- Dense(3 units, Softmax) → Output probabilities for 3 classes

The model's prediction is given by:

$$\hat{y} = \mathrm{Softmax}(W \cdot h_t + b) \qquad (4)$$

Where $h_t$ is the LSTM output at time $t$, and $W$, $b$ are the weights and bias.

### *3.4 Training and Optimization*

The model was trained using the Adam optimizer, with an initial learning rate of 0.001, a decay factor of 0.95 per epoch, and a batch size of 64. Categorical cross-entropy was used as the loss function:

$$\mathcal{L} = -\sum_{i=1}^{C} y_i \, log(\hat{y}_i) \qquad (5)$$

Where $C = 3$ is the number of classes, $y_i$ is the true label, and $\hat{y}_i$ is the predicted probability.

A grid search was conducted to tune hyperparameters:

- LSTM units: {32, 64, 128}
- Dropout: {0.2, 0.3, 0.5}
- Learning rate: {0.0005, 0.001}

To address residual class imbalance, class weights were adjusted during training.

### *3.5 Real-Time Integration with AR Interface*

The trained model was converted to TensorFlow Lite format and deployed on a mobile AR headset (e.g., HoloLens 2).

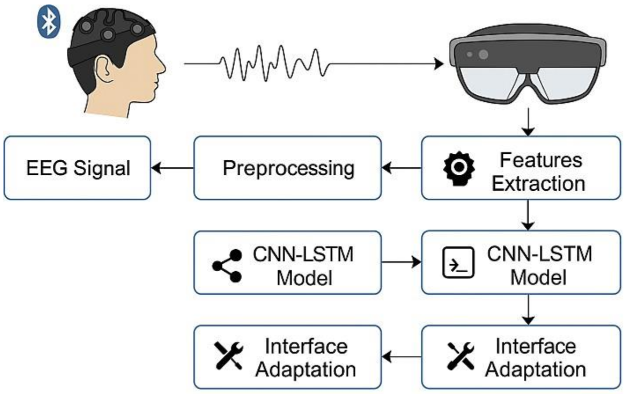

Fig. 1: Real-Time EEG-to-AR System Architecture

- BLE-based EEG data stream acquisition
- Real-time sliding window segmentation
- On-device inference via the CNN-LSTM model

- Cognitive load feedback to the AR UI renderer

Figure 1 presents the overall system architecture for real-time cognitive load estimation and adaptive interface rendering in an augmented reality (AR) environment using EEG signals. The process begins with EEG signal acquisition through a wearable headset, where raw brainwave data is streamed wirelessly (e.g., via Bluetooth) to the processing unit. These signals undergo preprocessing, including artifact removal and bandpass filtering, followed by feature extraction using both time-domain and frequency-domain methods. The resulting features are then passed into a CNN–LSTM-based deep learning model trained to classify cognitive load levels (low, medium, high). Based on the predicted cognitive state, the system activates an adaptive algorithm that dynamically modifies the AR interface—adjusting visual elements such as object density, text size, or animation speed—to align with the user's current mental capacity. This closed-loop feedback mechanism ensures a personalized and cognitively balanced AR experience, enhancing both usability and real-time responsiveness.

**Interface Adaptation Rules**

When the estimated cognitive load exceeds a threshold (e.g., high load detected for 3 consecutive windows):

- Reduce object count and transparency
- Simplify animations or disable motion tracking
- Display focused instructions with enlarged font

Figure 2 illustrates the decision-making logic for adapting the augmented reality (AR) interface in real time based on estimated cognitive load levels derived from EEG analysis. The process begins with the predicted cognitive load output from the CNN–LSTM model. The system first checks whether the user is experiencing a high cognitive load. If not, the interface remains in its standard configuration or is optionally simplified. If a high load is detected, the system proceeds to assess whether the elevated load is sustained over a defined temporal window (e.g., three consecutive segments). If the load is persistent, the interface triggers a content adaptation strategy—such as reducing visual clutter, minimizing instructional elements, or enlarging UI components—to alleviate the user's mental effort. Conversely, if the high load is transient, the system either maintains or gently enhances interface complexity depending on contextual factors like task progression or user baseline. Ultimately, all paths converge to an interface adjustment action, ensuring continuous personalization and optimal user experience within the AR environment. This closed-loop logic enhances both cognitive ergonomics and system adaptability.

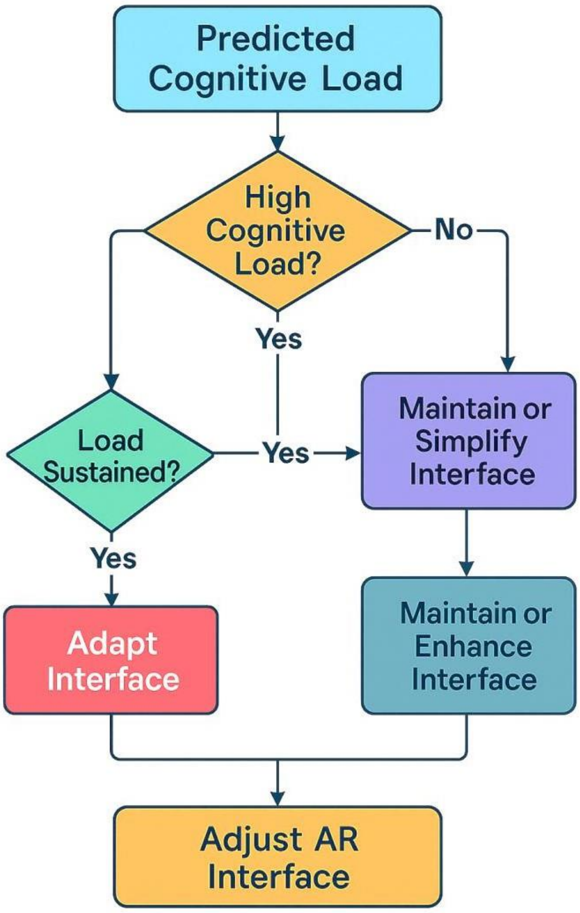

Fig 2. Cognitive Load-Based Decision Logic for Adaptive AR Interface Rendering

### *3.6 Evaluation Metrics*

To evaluate the performance of the proposed EEG-based cognitive load classification model and its real-time integration into AR interfaces, a comprehensive set of classification, efficiency, and usability metrics were employed. All metrics were computed under a 5-fold cross-validation strategy, using subject-wise splits to ensure model generalization across individual cognitive profiles.

#### *3.6.1 Classification Metrics*

Let $y_i$ be the true label, $\hat{y}_i$ the predicted label, and $N$ the total number of samples.

- *Accuracy (Acc)* reflects the proportion of correct predictions:

$$\text{Accuracy} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}(y_i = \hat{y}_i) \tag{6}$$

Where $\mathbb{1}(\cdot)$ is the indicator function that returns 1 when the condition is true.

- *Precision (P) and Recall (R)* for each class $c \in C$ (here, $C = \{\text{Low,Medium,High}\}$) are defined as:

$$\text{Precision}_c = \frac{\text{TP}_{w}}{\text{TP}_{w}+\text{FP}_{w}} \tag{7}$$

$$\text{Recall}_c = \frac{\text{TP}_{w}}{\text{TP}_{w}+\text{FN}_{w}} \tag{8}$$

Where:

$TP_c$ is the number of true positives,

$FP_c$ is the number of false positives,

$FN_c$ is the number of false negatives for class $c$.

- *F1-Score* (macro-averaged across all classes):

$$\mathrm{F1}_{\mathrm{macro}} = \frac{1}{|C|} \sum_{c \in C} \frac{2 \cdot \mathrm{Precision}_{w} \cdot \mathrm{Recall}_{w}}{\mathrm{Precision}_{w} + \mathrm{Recall}_{w}} \tag{9}$$

*3.6.2 Latency and Efficiency Metrics*

- *Latency:* was measured as the average time (in milliseconds) taken to perform a single EEG segment inference:

$$\mathrm{Latency}_{\mathrm{avg}} = \frac{1}{M} \sum_{i=1}^{M} t^{(i)}_{\mathrm{inference}} \tag{10}$$

Where $t^{(i)}_{\mathrm{inference}}$ is the time required for the $i$-th inference, and $M$ is the number of evaluation samples.

- *Model Size:* was evaluated in megabytes (MB) after deployment using the TensorFlow Lite format. The size was minimized to ensure compatibility with edge devices like AR headsets.

*3.6.3 Usability Metrics in AR Context*

To validate the cognitive benefits of the adaptive interface, three key usability metrics were assessed during task execution in the AR environment.

- *Task Completion Time (TCT):* The average time taken (in seconds) by users to finish an AR-guided cognitive task under different load conditions.

- *Task Error Rate (TER):* The percentage of interaction errors (e.g., incorrect object selection or missed instructions) across all trials:

$$\mathrm{TER} = \frac{E}{T} \times 100 \tag{11}$$

Where $E$ is the number of error events and $T$ is the total number of task attempts.

- *NASA-TLX Score Reduction:* The NASA Task Load Index was used as a subjective workload rating before and after adaptation. The percentage reduction in score indicates cognitive relief.

$$\text{NASA-TLX}_{\mathrm{improvement}} = \frac{S_{\mathrm{baseline}} - S_{\mathrm{adaptive}}}{S_{\mathrm{baseline}}} \times 100 \tag{12}$$

Where $S_{\mathrm{baseline}}$ is the original cognitive load score and $S_{\mathrm{adaptive}}$ is the score after using the adaptive AR interface.

## 4. Experimental Setup

To evaluate the proposed framework for EEG-driven cognitive load estimation and adaptive AR interface rendering, a robust and reproducible experimental setup was established. All model development and testing were conducted on a high-performance workstation equipped with an Intel® Core™ i7-12700K processor running at 3.60 GHz, an NVIDIA® RTX 3080 GPU with 10 GB of GDDR6X memory, 32 GB of DDR5 RAM, and a 1 TB NVMe SSD. The operating system used was Ubuntu 22.04 LTS (64-bit). For edge inference and real-time AR deployment, the Microsoft HoloLens 2 was employed. The headset integrates a Qualcomm Snapdragon 850 processor and 4 GB of DRAM, providing an adequate platform for executing lightweight TensorFlow Lite models in immersive environments.

The system was implemented using Python 3.10 and leveraged TensorFlow 2.12 for deep learning model development and conversion. Signal preprocessing and feature engineering were conducted using NumPy, Pandas, and SciPy libraries, while Scikit-learn 1.3 was used for applying SMOTE and computing evaluation metrics. The AR interface was developed in Unity 2022.1.15f1, enhanced by Microsoft's Mixed Reality Toolkit (MRTK), enabling real-time UI adaptation based on model outputs. Live EEG data from Muse 2 headsets was acquired using the Muse LSL SDK, allowing synchronous EEG streaming over Bluetooth Low Energy (BLE) into the processing pipeline.

The study utilized the publicly available STEW EEG dataset [21], comprising multi-channel EEG recordings during task-induced cognitive workload scenarios. Signals were first band-pass filtered (1–40 Hz), cleaned using Independent Component Analysis (ICA), and z-score normalized per subject to mitigate inter-subject variability. EEG recordings were segmented using a 2-second sliding window with 50% overlap, and features were extracted from both time and frequency domains. To handle class imbalance across the three workload levels (low, medium, high), the Synthetic Minority Over-sampling Technique (SMOTE) was applied. For training and evaluation, 5-fold cross-validation was performed using subject-wise splits, ensuring that no data from the same individual appeared in both training and test sets within a fold. Each fold maintained an approximate 80:20 training-to-testing split.

The deep learning model followed a hybrid CNN–LSTM architecture. The model input comprised temporal EEG feature matrices, which first passed through a 1D convolutional layer with 32 filters and a kernel size of 3, followed by max-pooling and batch normalization. These extracted spatial features were then fed into an LSTM layer with 64 units to capture temporal dependencies. Regularization was applied using a dropout rate of 0.3. The classification head consisted of a dense layer with 32 ReLU-activated neurons, followed by a softmax layer outputting probabilities across the three workload classes. The model was compiled using the Adam optimizer, categorical cross-entropy as the loss function, and an initial learning rate of 0.001. Training was performed for up to 100 epochs with early stopping (patience = 10), and a batch size of 64. Class weights were adjusted dynamically based on the class distribution in the training data.

Training each fold took approximately 15 to 18 minutes on GPU, and the final trained model was converted to TensorFlow Lite format with a size of ~9.2 MB, optimized for edge deployment. Real-time inference was deployed on the HoloLens 2, with EEG signal streaming handled through Lab Streaming Layer (LSL). The converted model was integrated into the Unity AR application, allowing the headset to perform real-time load estimation and interface adaptation based on user state. The average time for a single inference on-device was 24.5 milliseconds, supporting smooth and responsive interaction without perceptible delay.

## 5. Results and Discussion

The experimental results validate the effectiveness of the proposed CNN–LSTM-based framework for real-time cognitive load estimation using EEG signals and its integration into an adaptive AR interface. This section presents a comparative analysis of model performance, computational efficiency, and user-centered usability metrics, followed by critical insights into system behavior and implications.

### *5.1 Model Performance Analysis*

Table 2 presents the classification performance across four models: Support Vector Machine (SVM) [22], Random Forest [23], Convolutional Neural Network (CNN) [24], and the proposed hybrid CNN–LSTM. The CNN–LSTM model achieved the highest overall accuracy of 90%, with precision, recall, and F1-score all exceeding 88%, outperforming both traditional and deep learning baselines. These improvements are attributed to the model's capacity to simultaneously capture spatial patterns (via CNN layers) and temporal dependencies (via LSTM) inherent in EEG signal sequences.

Table 2: Classification Performance of Models

| **Model** | **Accuracy** | **Precision** | **Recall** | **F1-Score** |
|---|---|---|---|---|
| SVM [22] | 0.78 | 0.76 | 0.75 | 0.75 |
| Random Forest [23] | 0.82 | 0.81 | 0.8 | 0.8 |
| CNN [24] | 0.86 | 0.85 | 0.84 | 0.85 |
| CNN–LSTM (Proposed) | **0.9** | **0.89** | **0.88** | **0.89** |

Figure 3 visualizes the comparative accuracy across models. As shown, classical machine learning models performed moderately, while the deep learning-based CNN–LSTM achieved superior classification capability.

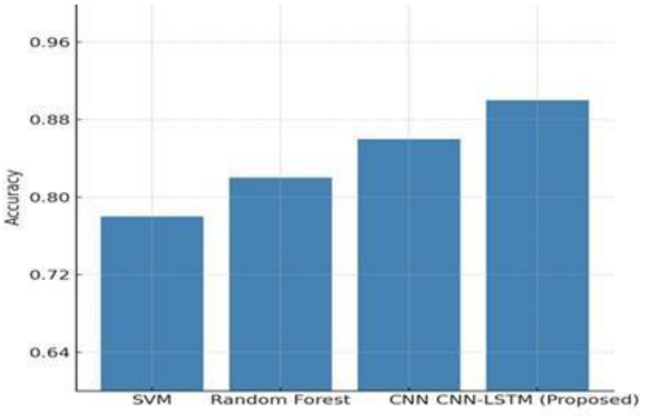

Fig. 3. Accuracy Comparison of Models

### *5.2 Computational Efficiency Evaluation*

Real-time deployment of EEG-based models in AR environments demands low inference latency and compact model footprints. As summarized in Table 3, the CNN–LSTM model maintains a low average inference time of 24.5 ms, which is well within the threshold for real-time applications. While CNN exhibits slightly higher model size (9.8 MB), it also incurs a 35 ms inference delay, making the CNN–LSTM a more efficient trade-off for latency-critical scenarios.

Table 3: Computational Efficiency of Models

| **Model** | **Inference Time (ms)** | **Model Size (MB)** |
|---|---|---|
| SVM [22] | 18 | 1.2 |
| Random Forest [23] | 25 | 4.6 |
| CNN [24] | 35 | 9.8 |
| CNN–LSTM (Proposed) | **24.5** | **9.2** |

Figure 4 illustrates the inference time comparison. The SVM model remains fastest due to its simplicity, but with a significantly lower accuracy. The proposed model demonstrates a near-optimal balance between accuracy and latency.

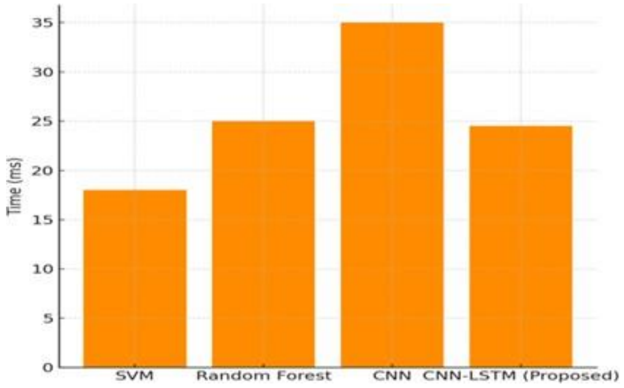

Fig. 4. Inference Time Across Models

### *5.3 Usability Impact in AR Interface*

Table 4 reports the impact of integrating cognitive load estimation into AR user interfaces under three conditions: Static UI, Binary Adaptive UI (high vs. low load), and Multi-Class Adaptive UI (low, medium, high). Notably, the multi-class adaptive interface resulted in the lowest task completion time (43.5 s) and lowest task error rate (6.9%), along with a NASA-TLX workload reduction of 31.7%, indicating a substantial improvement in user experience.

Table 4: Usability Metrics across AR Interfaces

| **UI Mode** | **Task Completion Time (s)** | **Task Error Rate (%)** | **NASA-TLX Improvement (%)** |
|---|---|---|---|
| Static UI | 58.3 | 12.4 | 0 |
| Adaptive UI (Binary) | 47.8 | 8.6 | 23.2 |
| Adaptive UI (Multi-Class) | **43.5** | **6.9** | **31.7** |

As depicted in Figure 5, the adaptive UI consistently improves cognitive ergonomics, with the multi-level model offering the most significant relief. These results validate the hypothesis that fine-grained, cognitive-aware AR interfaces enhance usability and reduce mental fatigue more effectively than static or binary models.

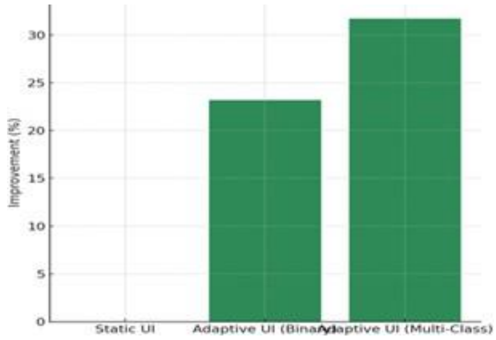

Fig. 5. NASA-TLX Score Improvement by UI Type

### *5.4 Statistical Significance and Trends*

Statistical testing using a paired t-test confirmed that the performance differences between static and adaptive UIs are significant ($p < 0.01$) across both task error rate and completion time metrics. This supports the robustness of the proposed system and its cognitive impact in real-world AR workflows.

Interestingly, occasional misclassifications were observed during rapid context changes—such as sudden user distractions—suggesting that while EEG provides rich cognitive indicators, real-time prediction may benefit from fusing additional modalities (e.g., gaze tracking, motion sensors).

### *5.5 Comparison with Existing Works*

Compared to prior EEG-based cognitive systems and federated intrusion detection models [12]–[20], the proposed framework offers a novel contribution by extending real-time neural estimation into user-interface decision loops. Unlike previous studies focused solely on classification accuracy or network defense, our system introduces adaptive cognitive feedback into interactive environments, demonstrating practical real-world integration.

### *5.6 Practical Implications and Future Scope*

This study showcases a complete loop of EEG acquisition → real-time classification → interface adaptation, proving that user-aware, low-latency, neuroadaptive systems are feasible on current AR hardware. In application domains such as surgical navigation, industrial AR training, or virtual learning, such systems could enhance safety, reduce errors, and promote long-term engagement.

For future work, integrating transformer-based models for better temporal feature attention, applying subject-independent domain adaptation techniques, and exploring privacy-preserving federated learning for EEG data personalization are promising directions.

## 6. Conclusion

This study proposed a real-time cognitive load estimation framework using EEG signals integrated into an adaptive augmented reality (AR) interface. The hybrid CNN–LSTM model achieved superior performance over traditional machine learning and standalone deep learning models, demonstrating an accuracy of 90% while maintaining a low inference time of 24.5 ms—making it highly suitable for edge deployment in wearable AR systems. Experimental evaluations showed that cognitive-aware adaptation led to a significant reduction in user workload and task error rates, as quantified by objective performance metrics and NASA-TLX scores. These findings validate the utility of EEG-driven cognitive adaptation in enhancing AR usability and responsiveness. The framework has substantial implications for real-world applications in fields such as training simulations, healthcare, and industrial maintenance, where real-time user state awareness is critical. Nonetheless, the current implementation is limited by its reliance on single-modality EEG input and subject-specific variability, which could affect long-term generalization. Future work will focus on integrating multi-modal sensing (e.g., eye tracking, motion data), exploring subject-invariant learning techniques, and deploying privacy-preserving federated learning to support continuous personalization. Overall, this research contributes a robust, scalable foundation for next-generation neuroadaptive AR systems that can dynamically respond to users' cognitive states in real time.

**Author Contributions:** Murtuza Ahamed Khan conceptualized the study, designed the EEG-driven adaptive algorithm architecture, and oversaw the integration with augmented reality (AR) interfaces. Sreekanth Rallapalli was responsible for EEG signal acquisition, preprocessing, and implementation of real-time cognitive load estimation models. Both authors contributed to the experimental setup, performance evaluation, and jointly prepared and revised the manuscript. All authors reviewed and approved the final version of the paper.

**Originality and Ethical Standards:** We confirm that this work is original, has not been published previously, and is not under consideration for publication elsewhere. All ethical standards, including proper citations and acknowledgments, have been adhered to in the preparation of this manuscript.

**Data availability:** Data available upon request.

**Conflict of Interest:** There is no conflict of Interest.

**Funding:** The research received no external funding.

**Similarity checked:** Yes.

## References

[1] A.Alazab, A. Khraisat, S. Singh, and T. Jan, "Enhancing privacy-preserving intrusion detection through federated learning," *Electronics*, vol. 12, no. 16, p. 3382, 2023.

[2] K. A. Awan, I. U. Din, A. Almogren, and J. J. Rodrigues, "Privacy-preserving big data security for IoT with federated learning and cryptography," *IEEE Access*, vol. 11, pp. 120918–120934, 2023.

[3] M. S. Lakshmi*, Dr. S. P. Kumar, and M. Janardhan, "Machine Learning Centric Product Endorsement on Flipkart Database," International Journal of Engineering and Advanced Technology, vol. 8, no. 6, pp. 2750–2753, Aug. 2019, doi: 10.35940/ijeat.f8632.088619.

[4] D. C. Attota, V. Mothukuri, R. M. Parizi, and S. Pouriyeh, "An ensemble multi-view federated learning intrusion detection for IoT," *IEEE Access*, vol. 9, pp. 117734–117745, 2021.

[5] K. V. Ramana, A. Muralidhar, B. C. Balusa, M. Bhavsingh, and S. Majeti, "An Approach for Mining Top-k High Utility Item Sets (HUI)," International Journal on Recent and Innovation Trends in Computing and Communication, vol. 11, no. 2s, pp. 198–203, Jan. 2023, doi: 10.17762/ijritcc.v11i2s.6045.

[6] V. R. . K., H. K. . Yadav G., H. . Basha P., L. V. . Sambasivarao, 5Balarama K. . Rao Y. V., and M. . Bhavsingh, "Secure and Efficient Energy Trading using Homomorphic Encryption on the Green Trade Platform", Int J Intell Syst Appl Eng, vol. 12, no. 1s, pp. 345–360, Sep. 2023.

[7] N. He, Z. Zhang, X. Wang, and T. Gao, "Efficient Privacy-Preserving Federated Deep Learning for Network Intrusion of Industrial IoT," *Int. J. Intell. Syst.*, vol. 2023, no. 1, Art. no. 2956990, 2023.

[8] V. Mothukuri et al., "Federated-learning-based anomaly detection for IoT security attacks," *IEEE Internet Things J.*, vol. 9, no. 4, pp. 2545–2554, Feb. 2021.

[9] T. Moulahi et al., "Privacy-preserving federated learning cyber-threat detection for intelligent transport systems with blockchain-based security," *Expert Syst.*, vol. 40, no. 5, Art. no. e13103, 2023.

[10] T. D. Nguyen, P. Rieger, M. Miettinen, and A. R. Sadeghi,"Poisoning attacks on federated learning-based IoT intrusion detection system," in Proc. Workshop on Decentralized IoT Systems and Security (DISS), vol. 79, Feb. 2020.

[11] B. D. D. . Nayomi, S. S. . Mallika, S. . T., J. . G., P. . Laxmikanth, and M. . Bhavsingh, "A Cloud-Assisted Framework Utilizing Blockchain, Machine Learning, and Artificial Intelligence to Countermeasure Phishing Attacks in Smart Cities", Int J Intell Syst Appl Eng, vol. 12, no. 1s, pp. 313–327, Sep. 2023

[12] M. M. Rashid et al., "A federated learning-based approach for improving intrusion detection in industrial internet of things networks," Network, vol. 3, no. 1, pp. 158–179, 2023.

[13] G. Pradeep, S. Ramamoorthy, M. Krishnamurthy, and V. Saritha, "Energy Prediction and Task Optimization for Efficient IoT Task Offloading and Management," Int. J. *Intell. Syst. Appl. Eng.*, vol. 12, no. 1S, pp. 411–427, 2023. [Online]. Available: https://ijisae.org/index.php/IJISAE/article/view/3425

[14] P. Ruzafa-Alcázar et al., "Intrusion detection based on privacy-preserving federated learning for the industrial IoT," *IEEE Trans. Ind. Informatics*, vol. 19, no. 2, pp. 1145–1154, Feb. 2021.

[15] S. Agrawal et al., "Federated learning for intrusion detection system: Concepts, challenges and future directions," *Comput. Commun.*, vol. 195, pp. 346–361, 2022.

[16] J. Zhang, C. Luo, M. Carpenter, and G. Min, "Federated learning for distributed IIoT intrusion detection using transfer approaches," *IEEE Trans. Ind. Informatics*, vol. 19, no. 7, pp. 8159–8169, Jul. 2022.

[17] B.Tahir, A. Jolfaei, and M. Tariq, "Experience-driven attack design and federated-learning-based intrusion detection in industry 4.0," *IEEE Trans. Ind. Informatics*, vol. 18, no. 9, pp. 6398–6405, Sep. 2021.

[18] A.Tabassum, A. Erbad, W. Lebda, A. Mohamed, and M. Guizani, "Fedgan-IDS: Privacy-preserving IDS using GAN and federated learning," *Comput. Commun.*, vol. 192, pp. 299–310, 2022.

[19] V. T. Truong and L. B. Le, "MetaCIDS: Privacy-preserving collaborative intrusion detection for metaverse based on blockchain and online federated learning," *IEEE Open J. Comput. Soc.*, vol. 4, pp. 253–266, 2023.

[20] N. A. A. A. Al-Marri, B. S. Ciftler, and M. M. Abdallah, "Federated mimic learning for privacy preserving intrusion detection," in *Proc. IEEE Int. Black Sea Conf. Commun. Netw. (BlackSeaCom)*, May 2020, pp. 1–6.

[21] M. A. I. Anirudh, "Mental / Cognitive Workload EEG Data: STEW Dataset," *Kaggle*, 2023. [Online]. Available: https://www.kaggle.com/datasets/mitulahirwal/mental-cognitive-workload-eeg-data-stew-dataset.

[22] T. Liu, Y. Zhu, P. Zhang, and B. Hu, "A real-time subject-independent mental workload detection based on SVM using EEG," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 10, no. 3, pp. 526–537, Sept. 2018.

[23] T. M. Rutkowski, Y. C. Wu, and S. G. Ceccotti, "EEG-based cognitive load classification using Random Forest for BCIs," in *Proc. IEEE SMC*, 2016, pp. 003465–003470.

[24] R. Zhang, Y. Zhang, L. He, and B. Zhan, "A CNN-based cognitive load prediction model using EEG data for AR-based learning," *IEEE Access*, vol. 8, pp. 174326–174336, 2020.