Robust Engagement Estimation in Mixed Reality Under Cognitive Attack

Peng Wu, Kaiming Huang, Mingyu Zhu, Zifan Zhou, Jianan Jiang, Nasim Ahmed, Md Mahedi Hassan, Shu Hong, Bin Li, Rifatul Islam, Tian Lan, Gang Tan, and Mahdi Imani

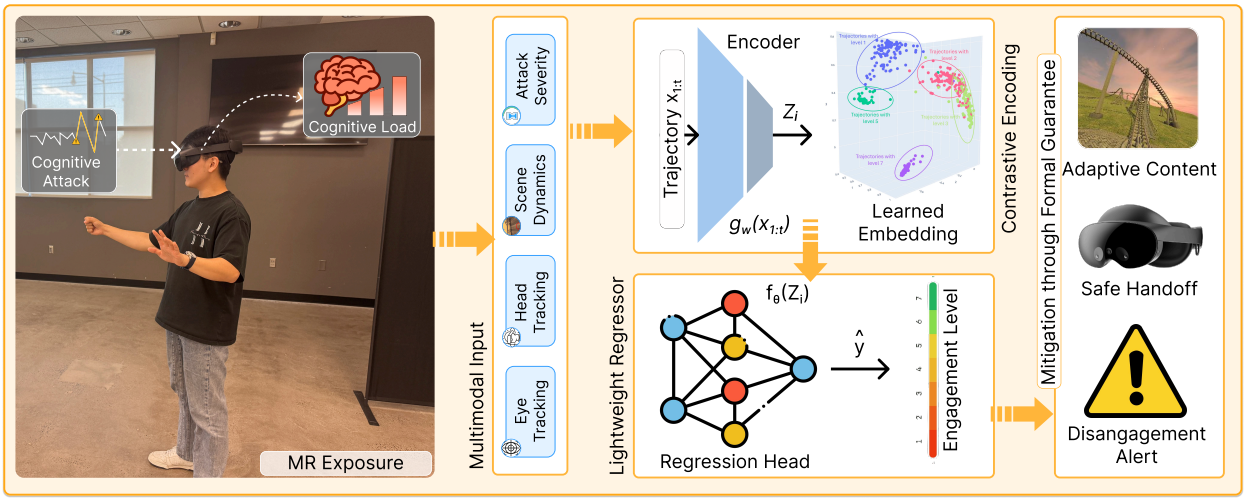

Figure 1: Our proposed two-stage framework estimates user engagement in Mixed Reality (MR) environments using multimodal time-series signals. Under cognitive attacks, instantiated here as latency perturbations in an MR task, the contrastive encoder learns invariant embeddings of behavioral trajectories that are mapped to a continuous engagement score via a calibrated regressor. As an initial step toward runtime assurance, we further analyze the learned model to derive *conservative safe-operating intervals* for controllable MR parameters.

Abstract—Engagement is a critical operational state in immersive systems, where users must sustain attention to virtual content despite distraction, workload, and degraded system responsiveness. In MR, one important source of disruption is a *cognitive attack*: an adversarial or system-level perturbation to the interaction loop that interferes with attention, situational awareness, or stable task execution. In this paper, we study induced latency as a concrete MR-relevant cognitive attack and present a two-stage model for estimating engagement from multimodal time-series data, including eye tracking, head motion, scene dynamics, and system latency. The first stage uses contrastive learning to encode variable-length behavioral trajectories into compact embeddings that suppress nuisance variation while preserving engagement-relevant structure. A lightweight regression head then maps these embeddings to a continuous engagement score. We evaluate the framework in a controlled hostage-rescue MR study with step-wise latency perturbations and minute-level engagement self-reports. Across cross-validation and held-out-user evaluation, the proposed model performs favorably relative to traditional regressors on pooled features, indicating that temporally structured multimodal representations provide a measurable benefit for engagement estimation under attack. We further show how formal verification can be used to derive conservative safe operating intervals for controllable MR parameters, connecting engagement estimation to runtime reasoning in safety-aware MR systems.

Index Terms—Mixed reality, engagement estimation, cognitive attacks, contrastive learning, safety-critical systems.

1 Introduction

Mixed Reality (MR) systems, spanning Virtual Reality (VR), Augmented Reality (AR), and their combinations, are increasingly deployed in critical domains such as pilot training, military operations, robotic surgery, manufacturing, and complex remote collaboration [6]. These systems blend real and virtual content into unified user experiences, relying on high-fidelity rendering, accurate tracking, and low-latency interaction to sustain immersion and performance. However, the reliability of MR in high-stakes environments hinges not only on system fidelity, but on the user's ability to maintain sustained cognitive engagement. Factors such as latency, jitter, resolution degradation, luminance shifts, and tracking artifacts can erode attention, degrade situational awareness, and compromise task success [8, 19].

Prior work has explored engagement estimation in VR and MR environments, typically as a predictive modeling task using behavioral or physiological signals [1, 7]. However, these models generally assume nominal conditions and do not account for the effects of system stres-

- *Peng Wu and Mahdi Imani are with Northeastern University, Boston, MA, USA. E-mail: wu.p@northeastern.edu, m.imani@northeastern.edu.*
- *Kaiming Huang, Mingyu Zhu, Zifan Zhou, Jianan Jiang, Bin Li, and Gang Tan are with The Pennsylvania State University, University Park, PA, USA. E-mail: kzh529@psu.edu, mintrrey@psu.edu, zqz5454@psu.edu, jnjiang@psu.edu, binli@psu.edu, gtan@psu.edu.*
- *Nasim Ahmed, Md Mahedi Hassan, and Rifatul Islam are with Kennesaw State University, GA, USA. E-mail: nahmed25@students.kennesaw.edu, mhassa27@students.kennesaw.edu, rislam11@kennesaw.edu.*
- *Shu Hong and Tian Lan are with George Washington University, Washington, DC, USA. E-mail: shu.hong@gwu.edu, tlan@email.gwu.edu.*

Manuscript received xx xxx. 202x; accepted xx xxx. 202x. Date of Publication xx xxx. 202x; date of current version xx xxx. 202x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.202x.xxxxxxx

sors. In contrast, this paper focuses on engagement under *cognitive attacks*, defined here as adversarial or system-level perturbations to the MR interaction loop that degrade attention, interaction quality, or situational awareness. We study latency perturbation as a concrete instance because it is both operationally relevant in MR and directly tied to rendering and interaction fidelity [3, 4]. Our objective is to model how engagement responds to these stressors so that MR systems can *anticipate disengagement early enough to act*. This dual function, engagement estimation and runtime assurance, is central to enabling adaptive, safety-aware MR.

We define engagement as a latent cognitive state reflecting sustained attention to task-relevant virtual content. It overlaps with, but is not identical to, related MR constructs: presence describes the feeling of "being there," immersion often refers to the technical and perceptual envelope of the system, and workload characterizes cognitive demand. Engagement is the construct of interest here because our goal is to estimate whether users remain attentive to MR-presented information when the system is perturbed. It is influenced by both *human-level features* (e.g., gaze patterns, head dynamics, prior MR familiarity) and *system-level parameters* (e.g., latency, resolution, luminance entropy), as well as their temporal evolution [36]. For example, brief latency spikes may not disrupt an experienced user, while sustained latency in a complex scene may cause rapid disengagement. This motivates a time-series modeling approach that jointly considers recent behavior and system dynamics. Modeling engagement under cognitive attacks presents a number of interrelated challenges that go beyond standard sequence prediction [21]. First, engagement supervision is both sparse and delayed: labels are typically sparse via self-report, which makes dense annotation infeasible and requires the model to infer latent cognitive states from weakly labeled behavioral sequences [30]. Second, user responses vary widely depending on prior experience, cognitive load, and attentional resilience, making it difficult to learn generalizable patterns across subjects [12]. Third, the input data are inherently multimodal and asynchronous, combining signals such as eye gaze, head motion, system latency, luminance, and scene entropy, each evolving at different temporal resolutions.

To address these challenges, we propose a two-phase learning framework that separates the task of temporal abstraction from that of engagement estimation. This decoupling improves generalization across users, scenes, and attack conditions while enabling efficient and interpretable inference.

- **Phase 1: Contrastive Embedding.** A recurrent encoder maps each variable-length multimodal trajectory to a fixed-dimensional vector. The encoder is trained using a soft InfoNCE objective guided by ordinal engagement similarity [17]: trajectories with similar self-reported engagement levels are pulled closer in the embedding space, while those with divergent levels are pushed apart. This training encourages the model to learn a cognitively meaningful latent space that captures engagement-relevant structure while suppressing nuisance variation such as user identity or scene layout.
- **Phase 2: Regression Head.** Once the encoder is trained, its parameters are frozen, and a lightweight feedforward head is trained to map embeddings to continuous engagement scores in $[1,6]$, followed by an optional stage that fine-tunes the encoder and head jointly at a low learning rate. Because the embeddings already reflect engagement semantics, the regression head remains simple and sample-efficient. To avoid prediction bias toward frequent mid-level scores, we use stratified sampling to ensure label diversity in each batch, so that rare but critical states such as full disengagement remain well represented.

Together, these two phases produce a model that supports real-time, cognitively grounded estimation of engagement under stress. The architecture is fast to compute, robust to user variability, and suitable for deployment in safety-critical MR systems that require continuous monitoring and early disengagement detection.

In addition to prediction, our framework supports formal verification of user engagement under disruptive conditions [29]. Specifically, we analyze whether ranges of controllable MR parameters such as latency or luminance are sufficient to keep predicted engagement above a desired threshold, yielding interpretable safety intervals that link rendering or interaction choices to cognitive outcomes. Although conservative, these intervals provide a principled basis for reasoning about safe operating regions in MR systems and make predictive guarantees formally actionable [9, 29].

To evaluate our framework, we conducted a controlled user study on a tactical MR testbed. Participants operated in a mixed-reality hostage-rescue scenario while step-wise latency conditions were applied once per minute to simulate cognitive attacks. Eleven users completed 10-minute sessions consisting of ten one-minute intervals, each ending with a self-reported engagement rating on a Likert-type scale [16]; the reported ratings span the operating range 1–6 that the model is trained and evaluated on. During each session, high-frequency multimodal data were captured, including gaze direction, head motion, system latency, luminance, scene entropy, and object interaction. Our approach achieves an average Mean Absolute Error (MAE) of 0.89 across four stratified folds and 0.84 on a held-out user set, demonstrating strong generalization and robustness. Visualization of the learned embedding space shows that engagement trajectories form distinct clusters by level, feature attribution analysis confirms that the model integrates both system and human signals rather than only attack intensity, and comparisons against Ridge, SVR, and XGBoost baselines show that the proposed architecture provides a consistent gain over common traditional regressors on pooled features. These results affirm the model's utility for cognitively grounded MR engagement monitoring under attack.

A large body of work has quantified how system latency degrades *objective* task performance in interactive and immersive systems (e.g., target acquisition, completion time, and error rates) [23, 25, 33]. Our study is complementary: rather than predicting an externally observable performance counter, we estimate the user's *subjective* engagement state under the same class of latency perturbations. Positioning latency as one instance of a cognitive attack lets us connect this objective–performance literature to a subjective, cognitively grounded target that prior latency research does not directly model.

Contributions. In summary, this paper makes the following contributions:

- A two-phase model for engagement estimation under cognitive attack that couples contrastive representation learning of variable-length multimodal MR trajectories with a lightweight regression head.
- An empirical evaluation on a controlled hostage-rescue MR study with step-wise latency perturbations, including label-level stratified cross-validation, a fully held-out-user generalization test, and comparisons against Ridge, SVR, and XGBoost baselines.
- An ablation that disentangles engagement from raw system performance, showing that attack intensity alone does not explain the model's predictions.
- A verification analysis that derives conservative safe-operating intervals for controllable MR parameters, together with measured inference cost supporting online monitoring.

2 Related Work

This section reviews the body of literature pertinent to engagement prediction in MR and VR environments, drawing from research in engagement recognition, time-series analysis, and contrastive learning.

2.1 Engagement Recognition in Immersive Environments

The study of user engagement is a well-established field in Human-Computer Interaction (HCI). Seminal work by [26] and [24] has explored the various facets of engagement in virtual environments, identifying key behavioral and physiological correlates. Engagement in educational settings, particularly in e-learning and virtual laboratories, has been extensively studied, with researchers focusing on how to measure and maintain student interest to improve learning outcomes

[11, 32]. In VR and MR, engagement is often discussed alongside presence, immersion, and workload, but these constructs are not interchangeable. Presence captures perceived "being there," immersion describes properties of the system and display, and workload reflects cognitive demand; engagement instead emphasizes whether users remain behaviorally and cognitively tuned to task-relevant MR content under changing conditions. Predicting engagement therefore often involves analyzing behaviors that indicate sustained attention, such as head and eye movements [2, 5, 34].

2.2 Time-Series Analysis for Behavioral Modeling

The sequential nature of the data in this project necessitates the use of advanced time-series analysis techniques. Machine learning methods for sequential data have been a subject of extensive research. Comprehensive reviews by [14] and [15] provide a thorough overview of both classical and deep learning approaches for time-series classification and regression. Models such as Long Short-Term Memory (LSTM) networks [10] and Transformers [31] have shown exceptional performance in capturing temporal dependencies in various domains. Furthermore, the analysis of behavioral time series, as explored by [38], provides a foundation for understanding how to model and interpret the complex patterns of human behavior over time.

2.3 Contrastive Learning for Time-Series Representation

Contrastive learning has emerged as a powerful self-supervised learning technique for learning effective data representations without relying on large amounts of labeled data [17]. This approach learns representations by maximizing the similarity between "positive" pairs (similar samples) and minimizing the similarity between "negative" pairs (dissimilar samples). While initially popularized in computer vision and Natural Language Processing (NLP), its application to time-series data is a rapidly growing area of research. Recent studies, such as the systematic review by [22] and the investigative work by [39], have demonstrated the potential of contrastive learning to learn robust representations for time-series forecasting and classification tasks. This project builds upon these advancements by adapting contrastive learning to the specific challenge of engagement prediction.

2.4 Verification Frameworks and Attack Mitigation in XR Systems

The use of formal verification for attack mitigation in XR systems has not yet been extensively explored, but it holds significant potential. XR applications are vulnerable to adversarial manipulations of visual input, latency, and rendering pipelines that can degrade user performance or cause cybersickness [13, 20]. While most existing work on verification has focused on vision or sequential models in general domains, frameworks such as DeepPoly [28] and Prover [27] demonstrate that it is possible to provide certified robustness against bounded perturbations. Translating these ideas to XR could allow systems to reason formally about whether changes to controllable MR parameters or to sensed behavior can push a user model outside acceptable operating conditions.

A recent study by Wu et al. [37] introduced a Bayesian Network framework for cybersickness prediction in VR, which not only models system and physiological factors but also supports formal probabilistic verification of safety thresholds. Similar approaches could be applied in XR to formally verify that parameter settings remain within ranges that prevent cognitive attacks and preserve user comfort. Additionally, studies in adversarial robustness of perception networks, such as Weng et al.'s work on certifying robustness against adversarial examples [35], provide the methodological building blocks. Applying such techniques to XR contexts would enable the derivation of interpretable safety intervals (e.g., allowable ranges of latency or luminance) that protect against cognitive attacks. Although this direction remains largely open, it points to a promising line of research at the intersection of XR, formal methods, adversarial ML, and human-in-the-loop safety.

3 Modeling Engagement Under Cognitive Attack: A Two-Phase Architecture

We introduce a two-phase learning framework for modeling user engagement in MR environments under cognitive attack, where induced latency disrupts attentional state and system responsiveness. The goal is to estimate engagement from multimodal time-series signals (eye tracking, head motion, scene descriptors, and system telemetry) in real time, even when cognitive stressors are present.

In our study design, users experienced fixed but varied levels of latency while performing the same scenario. After each minute, they reported their engagement level on a Likert-type scale (higher = greater engagement, lower = disengagement). The available information for modeling thus includes both *system-level features* (latency, frame rates, luminance statistics) and *human-level features* (gaze trajectories, head motion patterns). The same latency condition can have different effects across users: some may remain focused due to prior VR familiarity or resilient eye behavior, while others disengage quickly.

We begin by formalizing the available data and learning objective. In our study design, each participant is exposed to a fixed but varied level of cognitive attack over a continuous time interval. A cognitive attack is induced via a fixed latency condition (e.g., reduced frame rate), held constant over a window of length T. During this window, we record high-frequency multimodal signals, including system-level features such as frame latency, luminance statistics, entropy, and motion complexity from the scene, and human-level features including 3D gaze direction, head motion quaternions, and scene interaction focus. This results in a trajectory of the form $X = \{x_1, x_2, \ldots, x_T\}$, where each $x_t \in \mathbb{R}^D$ encodes the system and human signals at time t. After each interval, the user provides a scalar self-report $y \in \{1, \ldots, 6\}$ denoting their engagement level during that window, where $y = 1$ reflects complete disengagement and $y = 6$ indicates full cognitive immersion.

Thus, the full training dataset consists of N labeled sequences:

$$\mathscr{D} = \left\{ (X^{(i)}_{1:T^{(i)}}, y^{(i)}) \right\}_{i=1}^{N}$$

where $T^{(i)}$ is the length of the i-th trajectory and $y^{(i)}$ is the reported engagement score at the end of that trajectory. The goal is to estimate user engagement at inference time from partial or full observations:

$$\hat{y}_t = f(X_{1:t})$$

where $t < T$ ensues enabling early prediction and adaptive MR response. This setup reflects real-world deployment, where decisions must be made on streaming, truncated windows.

Two major challenges in building such models are:

- **Data sparsity and individual variability.** Engagement labels are minute-level, limited in number, and subject to user-specific resilience factors (e.g., gaming experience, VR familiarity, gaze stability). Collecting the massive datasets normally required for end-to-end time-series models is impractical.
- **Feature entanglement.** Raw multimodal features encode both *nuisance variation* (user identity, scene layout, background detail) and *true drivers of engagement* (interaction of human and system dynamics). A direct regression model risks conflating these factors, leading to poor generalization across users, scenes, or attack levels.

These challenges are further amplified by the high dimensionality and asynchronous nature of the input signals. Each user trajectory spans a variable-length time window and consists of multimodal features that evolve at different temporal granularities. Moreover, engagement labels are only available at coarse intervals, and identical latency levels can lead to vastly different outcomes depending on a user's internal state or behavioral patterns. In light of this, we aim to first learn a compact embedding that captures the latent engagement-relevant structure of each trajectory, without overfitting to superficial patterns like user identity or scene layout. This motivates a contrastive learning stage, in which trajectories are aligned according to engagement similarity in the absence of dense supervision.

To address these challenges, we propose a two-phase learning architecture that is both *sample-efficient* and *robust to behavioral variability*. Our model separates the task of **learning a temporally abstract, cognitively grounded representation** of user behavior from the task of **mapping that representation to engagement scores**. This decoupling has several benefits:

- The first phase learns a compact, low-dimensional embedding space where trajectories with similar engagement levels—even across different users, scenes, or attack intensities—are geometrically aligned. This embedding is trained contrastively to reflect latent cognitive state rather than surface-level differences.
- The second phase uses these embeddings to make predictions via an ordinal regression head. Because the embedding has already disentangled nuisance variation, the downstream mapping becomes simple, interpretable, and effective even with limited labeled data.

This structure, described below, enables the model to generalize across latency conditions, window lengths, and user profiles, while supporting real-time estimation, monitoring, and early detection of disengagement. The result is a deployable, cognitively informed inference system for adaptive MR applications.

3.1 Phase 1: Contrastive Engagement Embedding

We now describe the first stage of our model, which maps a multimodal behavioral trajectory into a fixed-size latent embedding that reflects engagement-relevant structure while abstracting away nuisance variation. This embedding space is trained using a contrastive objective, relying solely on coarse, sparse engagement labels provided at the end of each trajectory window.

Let $X_{1:t} = \{x_1, \ldots, x_t\}$ be a multimodal sequence of sensor features, where each $x_k \in \mathbb{R}^D$ includes eye tracking, head motion, scene dynamics, and system telemetry (e.g., latency). The engagement label $y_t \in \{1, \ldots, 6\}$ is reported once per window, after the user experiences the full cognitive attack condition. Our goal is to learn an embedding function

$$z = g_w(X_{1:t}) \in \mathbb{R}^E,$$

where g_w is an LSTM encoder parameterized by weights w, followed by masked average pooling and a projection layer. The output embedding z is L2-normalized to lie on the unit hypersphere (i.e., $\|z\|_2 = 1$), which improves numerical stability and enables cosine-based similarity learning. This training procedure does not require frame-level supervision. Instead, we exploit *ordinal relationships between trajectories*: those with similar engagement labels should produce similar embeddings, while those with divergent scores should be pushed apart in latent space.

Let $(X^{(i)}_{1:T^{(i)}}, y^{(i)})$ and $(X^{(j)}_{1:T^{(j)}}, y^{(j)})$ be two labeled trajectories. We define two forms of similarity:

(1) Engagement Similarity. Given the ordinal nature of the engagement labels, we define a soft similarity weight between label pairs as:

$$\omega_{ij} \propto \exp\left(-\frac{|y^{(i)} - y^{(j)}|}{\tau_s}\right),$$

where $\tau_s > 0$ is a temperature parameter controlling label sharpness. Higher ω_{ij} indicates stronger similarity between the perceived engagement levels of two trajectories. Note that this similarity is defined over labels, not the time-series themselves, allowing the model to generalize across users and window lengths.

(2) Embedding Similarity. Each trajectory is passed through the encoder to produce embeddings $z_i = g_w(X^{(i)}_{1:T})$ and $z_j = g_w(X^{(j)}_{1:T})$. The similarity between embeddings is computed using cosine similarity:

$$\mathrm{Sim}(z_i, z_j) = \frac{z_i^\top z_j}{\|z_i\|_2 \|z_j\|_2}.$$

Figure 2 illustrates the key contrastive alignment mechanism. For each pair of trajectories, we compute a soft engagement similarity based on label proximity and a cosine-based embedding similarity. The contrastive objective then aligns these two metrics, encouraging the learned embedding space to reflect engagement-level proximity.

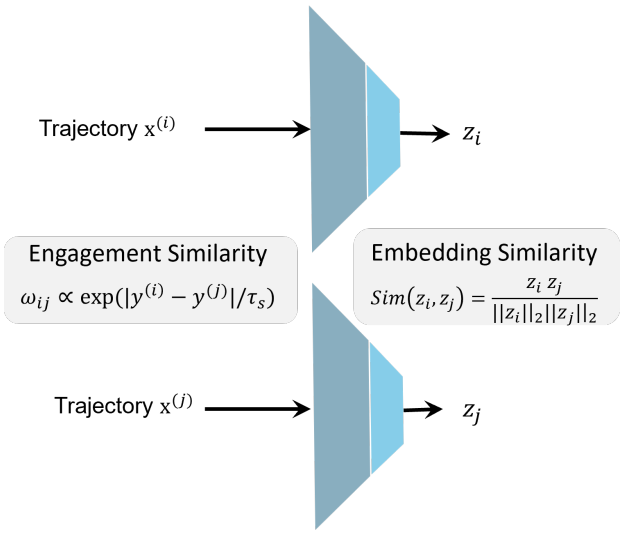

Figure 2: Illustration of the contrastive learning mechanism. Each trajectory is mapped to an embedding z_i, and the loss encourages alignment between label-derived engagement similarity ω_{ij} and cosine embedding similarity $\mathrm{Sim}(z_i, z_j)$.

Contrastive Loss. The core idea is to align the geometry of the embedding space with the structure of engagement similarity. The contrastive loss for anchor i is defined as:

$$\mathscr{L}_{\text{con}} = -\sum_{i=1}^{N} \omega_{ij} \log\left(\frac{\exp(\mathrm{Sim}(z_i, z_j)/\tau)}{\sum_{k \neq i} \exp(\mathrm{Sim}(z_i, z_k)/\tau)}\right),$$

where $\tau > 0$ is a separate temperature parameter for the embedding space. This soft InfoNCE formulation encourages embeddings from similarly engaged trajectories to lie closer together, regardless of user, scene, or attack condition. Thus, the parameters of the encoder w are optimized via standard gradient descent:

$$w \leftarrow w - \nabla_w \mathscr{L}_{\text{con}}.$$

This training strategy forces the LSTM to act as an engagement-sensitive filter: discarding irrelevant variation (e.g., background complexity, scene layout, user identity), while preserving latent cognitive dynamics. It also enables alignment of trajectories of different lengths into a common static embedding space.

The clustering effect encouraged by this contrastive loss—trajectories grouping by engagement level—is confirmed empirically on the learned embeddings via the t-SNE projection in Fig. 5. The resulting embedding then serves as input to the downstream prediction head (Phase 2), described next.

3.2 Phase 2: Regression from Embedding

Once the encoder g_w is trained to produce engagement-sensitive embeddings, we freeze its parameters and train a lightweight regression head $f_\theta : \mathbb{R}^E \to \mathbb{R}$ that maps each static embedding z to a continuous engagement score $\hat{y} \in [1, 6]$. While engagement labels are ordinal in nature, we treat this as a regression problem to directly optimize numerical prediction accuracy while keeping the head architecture simple and fast.

Model structure. The prediction head is a two-layer feedforward network parameterized by θ. Given an input embedding $z \in \mathbb{R}^E$, the model computes:

$$\hat{y} = f_\theta(z) = W_2 \cdot \mathrm{ReLU}(W_1 z + b_1) + b_2,$$

where $\{W_1, W_2, b_1, b_2\} \subset \theta$. The output $\hat{y} \in \mathbb{R}$ is interpreted as the predicted engagement level.

For a given trajectory $X_{1:t}^{(i)}$ with label $y^{(i)}$, the encoder first computes the embedding $z_i = g_w(X_{1:t}^{(i)})$, which is then passed to the regression head. The primary training objective is to minimize mean squared error (MSE) between predictions and true labels:

$$\mathscr{L}_{\text{reg}} = \frac{1}{N} \sum_{i=1}^{N} \left(f_\theta(z_i) - y^{(i)} \right)^2 .$$

To improve robustness across the full range of engagement levels, we adopt a stratified sampling strategy during training so that each mini-batch includes trajectories from across the engagement spectrum. This mitigates overfitting to mid-range labels and improves robustness to data imbalance.

Real-time prediction. At inference time, the model operates over a rolling window of streaming data. Given a time-series window $X_{1:t}$, the encoder produces an updated embedding $z_t = g_w(X_{1:t})$, which is passed through the regression head to obtain $\hat{y}_t = f_\theta(z_t)$. To support applications that rely on discrete engagement levels, the model's continuous output can optionally be rounded to the nearest integer:

$$\hat{y}_t^{\text{int}} = \text{Round}\,(\hat{y}_t)\,, \quad \hat{y}_t^{\text{int}} \in \{1, \ldots, 6\}.$$

This provides a categorical prediction that aligns with the user-reported Likert-style scale, suitable for downstream thresholding, binning, or visualization.

The two-phase training process is shown in Figure 3. In Phase 1, the encoder g_w is trained using a contrastive loss that aligns embeddings based on engagement label proximity. Each mini-batch consists of multimodal trajectories and their corresponding engagement labels, from which soft label-derived weights ω_{ij} are computed. These weights guide the InfoNCE-style objective, pulling together embeddings of similar engagement and pushing apart those with dissimilar levels. Once trained, the encoder is frozen, and Phase 2 proceeds to fit a lightweight regression head f_θ, which maps the embeddings to continuous engagement scores. The regression loss is the mean squared error between predicted and reported engagement scores. Finally, we apply an optional fine-tuning stage: after the regression head has converged with the encoder frozen, we unfreeze the encoder and jointly optimize the full model (encoder and head) end-to-end at a low learning rate. This lets the encoder make small task-specific adjustments while preserving the structure learned during contrastive pretraining. All results reported in this paper include this fine-tuning stage; the epochs and learning rates for all three stages are listed in Table 1.

Figure 3 summarizes our full architecture, highlighting the contrastive encoder that abstracts temporally rich behavior into a static embedding, and the regression head that translates this representation into continuous engagement estimates. This decoupled, two-stage procedure enables sample-efficient learning and robust generalization across users, latency conditions, and variable-length trajectories.

4 Verification for the Model

To complement the predictive evaluation, we also investigated whether the engagement model can be verified with formal guarantees under cognitive attacks. We use *Prover* [27] as the verifier; it relies on *DeepPoly*-style relational abstractions [28] to propagate linear constraints through the unrolled LSTM and its nonlinear gates. These tools are well-established in the verification community and let us pose guarantees over *bounded ranges of system parameters*.

From robustness certification to safety intervals. Prover was originally developed to certify the *robustness* of recurrent networks, such as LSTMs, against bounded input perturbations [27]. Given an input region defined by an ℓ_p-ball or other bounded set, the tool proves that the network's output does not change prediction across all inputs in that region. This guarantees stability of the model under adversarial perturbations.

For our purpose, we reinterpret this robustness framework as a way to verify *safety intervals* for mission-related system parameters. Suppose we want to check whether a given interval I of system parameters (e.g., latency between 15–35 ms) is sufficient to keep engagement above a target threshold τ. The safety requirement can then be expressed as

$$\forall x \in I : \; f(x) \;\geq\; \tau,$$

where $f(x)$ is the predicted engagement score. In other words, every input generated under the parameter constraints in I must yield an engagement level at least τ. If Prover can discharge this property on the unrolled LSTM, then I is formally verified as a safe operating interval for threshold τ. We focus on thresholds $\tau = 4$ and $\tau = 5$ because they align with natural breakpoints in the engagement scale: a score of 4 marks the midpoint boundary between distracted and attentive states, while a score of 5, the second highest observed score, represents strong engagement. Verifying guarantees at these two levels therefore captures both an acceptable level and a higher standard of engagement.

5 Dataset collection and process

The data pipeline is designed with a high degree of automation and flexibility, accommodating multiple data sources and formats to streamline the experimental process.

5.1 Data introduction

We conducted a controlled MR user study on a tactical search-and-rescue testbed using a Microsoft HoloLens 2 headset in a consistent room-scale setup across all participants. The cohort comprised 11 male college students aged 25–32. Each participant completed a single ten-minute session in a mixed-reality hostage-rescue scenario containing hostages, enemies, fires, medical kits, and weapons. Participants were free to move, inspect the scene, and optionally interact with virtual elements using gaze or controller input, but no explicit score or time-critical task objective was imposed. This open-ended design was intended to elicit engagement with the MR interface itself rather than optimize a narrow task-performance metric.

To modulate cognitive stress, we injected step-wise latency cognitive attacks once per minute with three broad ranges: low latency (45–60 FPS), medium latency (20–45 FPS), and high latency (5–20 FPS). These conditions were pseudo-randomized across the ten one-minute intervals so that each participant experienced multiple attack intensities during the session. At the end of each minute, an in-headset prompt asked participants to rate their level of engagement with the mixed reality display during the last minute. Here, 1 denoted complete disengagement from the MR content and higher values denoted stronger engagement; the ratings used for modeling span the range 1–6. The next latency level was applied immediately after the participant submitted the rating.

Throughout each session, we continuously captured multimodal telemetry reflecting both user behavior and scene dynamics. Human-centric signals included binocular eye tracking (3D eye positions, gaze directions, and ocular rotations), gaze-target encodings that identify what category of object the user was attending to, and 6-DoF head pose (3D position and rotation quaternions). System- and scene-centric signals included a perturbation indicator capturing the intensity of each one-minute latency attack together with its start timestamp, plus nine video-derived descriptors that characterize scene motion and appearance: optical flow magnitude, histogram-of-oriented-gradients features, edge intensity, temporal smoothness, brightness flicker, spectral entropy, spatial frequency, luminance, and contrast. These streams were sampled at their native rates and later aligned to the minute-wise engagement labels by matching each label to the nearest telemetry timestamp.

This acquisition protocol yields short, multi-sensor sequences paired with scalar engagement labels, enabling supervised learning with sequence encoders. The eye and head signals capture overt attentional behavior, the scene descriptors summarize visual complexity and stability, and the latency channel encodes the cognitive attack itself. As shown in Fig. 4, the label distribution is uneven across both levels and individuals: some participants did not traverse all engagement levels

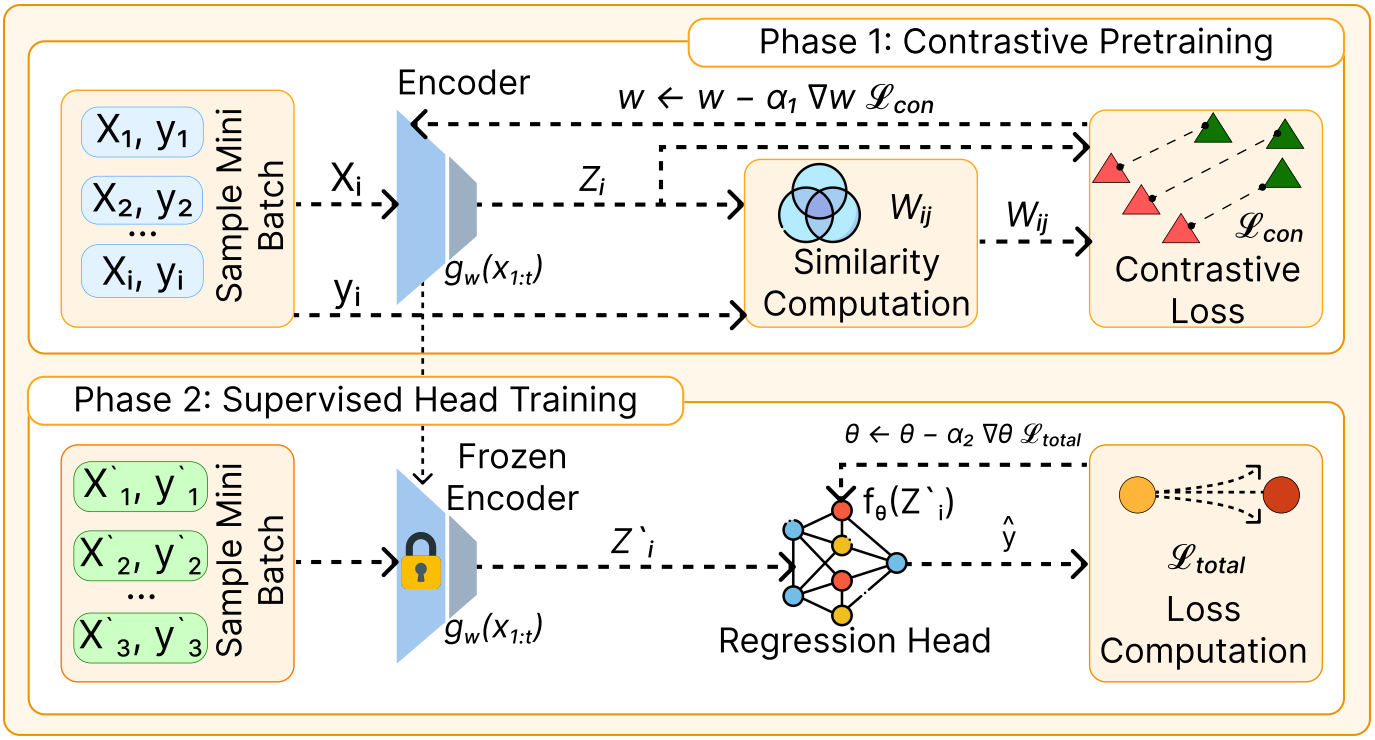

Figure 3: Two-phase training framework for engagement prediction. Phase 1 performs contrastive pretraining of the encoder using similarity-based objectives. Phase 2 freezes the encoder and trains a regression head with a supervised mean-squared-error loss.

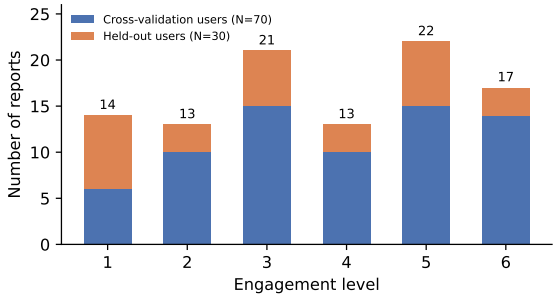

Figure 4: Distribution of engagement labels used in the study, split into the eight cross-validation users ($N = 70$ reports) and the three held-out users ($N = 30$ reports). Each bar is annotated with the total number of reports at that engagement level. The labels span the operating range 1–6.

during the protocol, and several contributed only three distinct levels. This heterogeneity motivates modeling and evaluation procedures that are robust to sparse and imbalanced labels.

5.2 Data Processing Pipeline

The raw data undergoes a systematic, multi-step processing pipeline to prepare it for the machine learning models. This pipeline includes the following stages. Design choices mirror the constraints of real-time deployment (low latency, variable window lengths) and the study protocol (minute-wise labels):

1. **Timestamp Alignment**: For each user, the engagement scores (labels) are loaded from their corresponding files. The timestamps of these scores are then precisely matched to the closest timestamps in the aligned feature data. This critical step ensures that each label is correctly associated with a specific point in the time-series data.
2. **Feature Preprocessing**: Two optional preprocessing steps can be applied to enhance the quality of the feature data:
   - **Rolling Average**: A rolling mean can be applied to the feature data over a specified window. This helps to smooth the data and reduce noise.
   - **Subsampling**: The data can be subsampled by taking every n-th data point. This is useful for reducing sequence length and redundancy, which can improve training efficiency.
3. **Sample Creation**: For each engagement score, multiple training samples are generated. Each sample consists of a sequence of feature vectors that lead up to the time of the engagement score. A `max_lookback` parameter defines the maximum length of these sequences. To augment the dataset and improve model robustness, the `num_samples_per_score` parameter can be set to create multiple sequences for each score. Each of these sequences is generated with a randomly chosen start point within the look-back window, ensuring that the model is exposed to sequences of varying lengths and that no two samples are identical.
4. **Padding and Masking**: Since the generated sequences can have variable lengths (up to `max_lookback`), they are front-padded with zeros to create batches of uniform length. This is a necessary step for efficient batch processing in deep learning models. A corresponding binary mask is also generated to distinguish the real data points from the padded values. The models use this mask to ignore the padded values during computations, ensuring that the results are not affected by the padding.

The final output of this comprehensive data pipeline is a set of `(sequence, mask, label)` tuples, which are ready for use in training and evaluating the models.

5.3 Training

Training follows the two-phase strategy outlined in the methodology. All hyperparameters for data processing, model architecture, and optimization are explicitly configured and recorded to ensure reproducibility. We use the Adam optimizer [18] in both phases due to its efficiency

and convergence stability. The contrastive pretraining phase optimizes a continuous contrastive loss, while the regression phase minimizes Mean Squared Error (MSE). After each phase, the best-performing model is selected based on validation performance to enable fair comparison and subsequent evaluation.

Table 1 lists the full architecture and hyperparameter configuration used for every reported result. The encoder is a two-layer LSTM (hidden sizes $128 \rightarrow 64$) followed by a projection to a 16-dimensional, L2-normalized embedding; the regression head is a single-hidden-layer MLP ($16 \rightarrow 64 \rightarrow 1$). All variants are trained with the identical configuration, and predictions are constrained to the empirical engagement range $[1, 6]$ at inference time.

5.4 Evaluation and Cross-Validation

We use **label-level stratified 4-fold cross-validation**, which balances the engagement label distribution across folds to mitigate class-imbalance artifacts (e.g., rare levels missing from a fold). This split strategy encourages learning class-specific patterns rather than overfitting to a specific user's distribution. For each fold, the complete two-phase training process is executed from scratch. We report MAE and standard deviations across folds. Here, label-level means we stratify individual engagement labels (rounded to 1–6) across four folds while ignoring user boundaries, and all samples derived from the same label remain within a single fold to avoid leakage.

Concretely, the four folds are generated with scikit-learn's `StratifiedKFold` (random seed 422) over the eight cross-validation participants (users 1, 4, 5, 6, 7, 9, 10, and 11); because stratification is over engagement labels rather than users, samples from a given participant may appear in different folds. We also conduct a **fully held-out generalization test** on the three remaining users (users 2, 3, and 8), whose data are excluded from both training and cross-validation. This evaluates performance on truly unseen individuals, a deployment-relevant condition for MR systems.

6 Results and Discussion

To evaluate the effectiveness of the proposed two-phase framework, we conducted label-level stratified four-fold cross-validation using an LSTM encoder. Performance is reported with MAE between the predicted and self-reported engagement scores (lower is better).

6.1 Cross-Validation Performance

Table 2 provides a detailed breakdown of performance across four cross-validation folds, aligned with the latest analysis.

The model achieves an average MAE of 0.89, which is practically meaningful given the inherent subjectivity of self-reported engagement scores. An error of less than one level is reasonable because adjacent ratings on a 1–6 scale are semantically close and often noisy even for the same user. The variance across folds, highlighted by the superior performance in Fold 3 (MAE 0.693), points to data sparsity in certain label regimes, a common challenge in behavioral studies. Despite this, the model remains stable across folds, and the average error for most engagement levels stays around or below the 1.0 threshold.

6.2 Baseline Comparison

To evaluate whether the proposed architecture is necessary, we compared it against three traditional non-sequential regressors: Ridge, SVR, and XGBoost. For each trajectory, we temporally averaged the variable-length multimodal sequence into a single fixed-length feature vector and trained each regressor on the same label-level stratified 4-fold splits used by the proposed model. Table 3 shows a clear spread in performance: Ridge reaches 1.09 ± 0.18 MAE, SVR reaches 1.24 ± 0.15 MAE, and XGBoost is the strongest traditional baseline at 0.96 ± 0.12 MAE, while the proposed two-stage model achieves 0.89 ± 0.19 MAE. This indicates that pooled multimodal features already carry useful predictive signal, but the full model remains the most accurate overall while additionally producing structured embeddings that support trajectory-length analysis, visualization, and formal verification.

6.3 Generalization Test (Held-Out Users)

We evaluate generalization on a fully held-out set of three users excluded from training and cross-validation. The model achieves an overall MAE of 0.84 on 1,200 holdout samples, indicating strong performance on unseen users (Table 4).

The model demonstrates strong generalization to unseen users, achieving an overall MAE of 0.84, which is better than the cross-validation average. This suggests that the features learned by the model are not merely memorizing the training participants. The performance is consistently high across all three held-out users, with MAE ranging from 0.66 to 0.95. This behavior is important for MR deployment scenarios where per-user calibration may be limited or unavailable.

6.4 Inference Cost

To verify real-time feasibility, we benchmarked the deployed engagement-prediction pipeline on a local server (Intel i9-12900H CPU, 16 GB RAM, NVIDIA RTX 3070 Laptop GPU, 8 GB VRAM). The system samples one video frame every 0.5 s and forms a fixed input window of the most recent 60 feature rows. We report two pure-compute metrics, excluding file transport, server polling, and response waiting, measured over 100 requests after 20 warm-up runs. The *model inference latency* (prepared input to engagement output) averages 13.2 ms (std 7.0 ms; range 8.6–65.6 ms), while the *end-to-end computation time* (from video-frame processing to engagement output, including feature extraction and inference) averages 527.8 ms (std 56.2 ms; range 436–712 ms). The model forward pass itself is therefore negligible, and even the full pipeline including video feature extraction completes in roughly half a second per update—far below the cadence required for engagement monitoring, whose self-reports are minute-level. This confirms that the architecture is compatible with real-time deployment in adaptive MR systems; the dominant cost is video feature extraction rather than the model, making it the natural target for further optimization.

6.5 Embedding and Trajectory-Length Analyses

Further analysis confirms the model's robustness and provides insights for practical deployment. The t-SNE visualization of the learned embeddings (Figure 5) reveals a highly structured representation space where different engagement levels are effectively separated. This distinct clustering, especially for extreme engagement states, confirms the success of the contrastive pre-training phase in learning meaningful features.

To understand the impact of the look-back window on performance, we analyzed the prediction error (MAE) across various trajectory lengths. As shown in Figure 6, our analysis shows that the model's accuracy improves with more context and stabilizes with input windows of around 30 seconds. This finding is critical for real-time applications, as it demonstrates that high accuracy can be achieved with short, low-latency input windows. The combination of a well-structured embedding space and strong performance with short trajectories makes the model highly suitable for efficient, real-time engagement tracking in MR systems.

6.6 Feature Importance Analysis

To understand which signals the model relies on, we compute permutation-based feature importance. After training, we independently shuffle each feature in the test set and measure the resulting increase in MAE; larger increases indicate greater importance. As shown in Table 5, the attack indicator (`AttackIntensity`) is most influential, but several non-attack cues also contribute substantially, including scene descriptors (spatial frequency, temporal smoothness, luminance) and gaze features. This indicates the model leverages diverse behavioral and scene signals rather than overfitting to the attack channel alone. For MR system design, this is important: the model is sensitive not only to the injected perturbation itself, but also to how the user visually and behaviorally responds to that perturbation in context.

Table 1: Model architecture and training hyperparameters used for all reported results.

Architecture		**Training and optimization**	
Encoder (LSTM)		*Contrastive pretraining (Phase 1)*	
Input feature dimension	40	Epochs / learning rate	300 / 5×10^{-4}
Stacked LSTM hidden sizes	$128 \rightarrow 64$	Label τ_s / embed. τ / similarity	0.1 / 0.05 / cosine
Projection ($64 \rightarrow 24 \rightarrow 16$)	LayerNorm, ReLU, drop. 0.3	*Regression (Phase 2) & fine-tuning*	
Embedding size E (L2-norm.)	16	Epochs / learning rate / loss	100 / 10^{-3} / MSE
Regression head		Fine-tune epochs / learning rate	100 / 10^{-4}
Hidden / output ($16 \rightarrow 64 \rightarrow 1$)	ReLU, dropout 0.2	*Optimization & data processing*	
		Optimizer / batch size	Adam / 64
		Rolling window / subsample step	60 / 40
		Max look-back / samples per score	60 / 40

Table 2: Performance metrics per fold for the LSTM encoder. The rightmost column reports the mean and population standard deviation across folds.

Metric	**Fold 1**	**Fold 2**	**Fold 3**	**Fold 4**	**Average**
Overall MAE	0.883	1.189	**0.693**	0.776	**0.89 ± 0.19**
MAE Level 1	0.935	3.000	**0.469**	0.999	1.35 ± 0.97
MAE Level 2	0.737	0.753	**0.411**	0.509	0.60 ± 0.15
MAE Level 3	1.481	0.945	**0.523**	0.750	0.93 ± 0.35
MAE Level 4	1.446	**0.625**	0.943	0.776	0.95 ± 0.31
MAE Level 5	0.483	1.613	**0.214**	1.056	0.84 ± 0.54
MAE Level 6	0.635	0.819	1.570	**0.539**	0.89 ± 0.41

Table 3: Cross-validation comparison between the proposed model and traditional regressors on temporally averaged trajectory features.

Model	**Feature Setting**	**CV MAE**
Ridge baseline	Temporal-mean feats.	1.09 ± 0.18
SVR baseline	Temporal-mean feats.	1.24 ± 0.15
XGBoost baseline	Temporal-mean feats.	0.96 ± 0.12
Proposed two-stage	Contrastive LSTM	0.89 ± 0.19

Table 4: Performance on a holdout test set of three unseen users.

Holdout User	**Mean Absolute Error (MAE)**
New User 1	0.66
New User 2	0.92
New User 3	0.95
Overall Performance	**0.84**

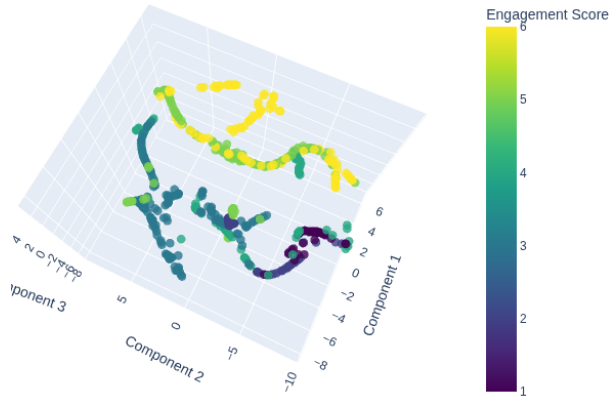

Figure 5: t-SNE projection of the 16-D embeddings produced by the LSTM encoder for all test sequences in Fold 3. Colours denote the ground-truth engagement score (1–6). Well-separated clusters at the extremes (dark purple, yellow) corroborate the quantitative results.

6.7 Disentangling Engagement from System Performance

A natural concern is that the model may estimate engagement primarily from the injected latency (`AttackIntensity`) rather than from user behavior, i.e., that it reduces to a latency-to-engagement lookup. To test this directly, we retrain the full two-phase model under three feature regimes using the identical label-level stratified 4-fold protocol: (a) the *full* model with all 40 features; (b) an *attack-only* model that receives `AttackIntensity` as its single input; and (c) an *attack-excluded* model trained on the remaining 39 behavioral and scene features. Table 6 reports the results.

Two findings argue against the latency-lookup hypothesis. First, the attack-only model is the weakest of the three (MAE 1.07), substantially worse than the full model (MAE 0.89): attack intensity alone is not sufficient to recover engagement. Second, removing the attack channel entirely degrades the full model only modestly (MAE $0.89 \rightarrow 1.04$), and the attack-excluded model still outperforms the Ridge (1.09) and SVR (1.24) baselines that retain access to all features including `AttackIntensity`. The encoder has therefore internalized behavioral and scene correlates of engagement that survive removal of the explicit attack signal. Both channels contribute, but neither alone explains the model's predictions—consistent with the permutation-importance analysis above, in which the aggregate importance of non-attack features exceeds that of `AttackIntensity` alone.

6.8 Verification results and analysis

Table 7 reports the verified parameter intervals obtained using our verification methodology described in Section 4.

The results in Table 7 show that the approach using Prover can indeed verify safe operating intervals for the engagement model. However, the verified ranges are extremely tight; for example, luminance is confined to less than a single unit. This outcome is not surprising given the structure of the model: it combines 40 input features at each timestep, spanning both system-level parameters (e.g., latency, frame rate, luminance) and user-level features (e.g., gaze stability, blink frequency, head motion). When verifying a property, Prover must account for all possible variations of the uncontrolled user features while simultaneously constraining the system parameters. To ensure that the engagement score remains above the threshold for *all* inputs consistent with a chosen interval, the verifier produces the interval to a very conservative slice of the input space. These guarantees are therefore sound but narrow, reflecting the worst-case combination of stochastic user behavior and adversarial perturbations of controllable system parameters. As a result, while the verified intervals provide a rigorous baseline, their small size limits their direct applicability in realistic MR deployments. Even so, they demonstrate that an engagement model can be connected back to controllable MR parameters such as latency, luminance, and scene motion, rather than treated as a purely black-box predictor.

More broadly, robustness verification as performed by Prover is inherently deterministic: it proves that outputs are stable across a bounded region of inputs. Engagement, in contrast, is a stochastic cognitive state influenced by human variability and environmental noise. Thus, even if Prover verifies that engagement $\geq \tau$ for all inputs in a region, this does not quantify the probability that real users will maintain that

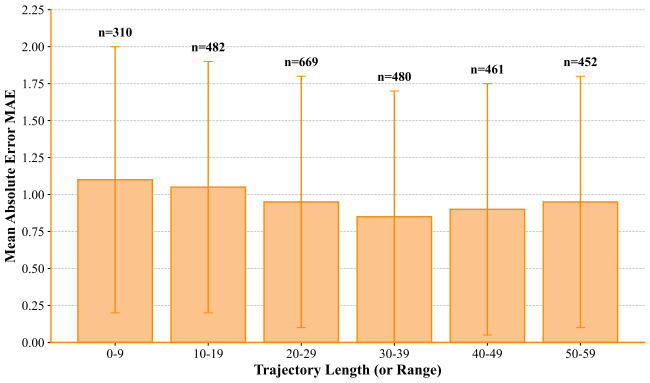

Figure 6: Relationship between input trajectory length and prediction error (MAE) for the best-performing LSTM model. Error bars indicate one standard deviation; n above each bar denotes the number of test samples in the bucket.

Table 5: Top 10 most influential features for the LSTM encoder, ranked by permutation importance. The importance score represents the increase in MAE when the feature is permuted.

Feature	Importance (MAE Increase)
AttackIntensity	1.0324
Feature_spatial_frequency	0.4454
Feature_temporal_smoothness	0.4145
RightGazeDirectionZ	0.1865
RightGazeDirectionY	0.1688
Feature_luminance	0.1216
ObjectCount_encoded	0.0328
Feature_spectral_entropy	0.0057
HeadRotationY	0.0055
LeftEyePositionZ	0.0045

engagement level. For deployment and enforcement, however, it is often more practical to provide *probabilistic guarantees*, where the system ensures that engagement stays above the threshold with high probability and confidence rather than under every conceivable input variation. Such guarantees yield broader, more interpretable operating ranges that can still be enforced meaningfully in practice, even if they admit a small residual risk. Bridging this gap requires methods that combine symbolic verification with probabilistic reasoning, so that verified intervals can be interpreted as actionable MR operating conditions with statistical confidence. This motivates future directions such as integrating probabilistic analysis with abstract domains, tailoring abstractions to multimodal and asynchronous features, and exploring hybrid symbolic–statistical approaches that yield guarantees aligned with deployment needs.

7 Limitations and Future Work

Our results are encouraging, but several limitations should shape their interpretation.

Dataset Expansion and Diversity. The current dataset, while richly detailed, is modest in size. The subjectivity of minute-level self-reported engagement and the natural imbalance in label distributions present challenges. The cohort also consisted of 11 male college students aged 25–32, which limits claims about population-level generalization across gender, age, and background. At the same time, this relatively homogeneous cohort is still useful for an initial proof-of-concept study because it reduces between-subject variance and allows us to isolate how engagement changes under controlled latency perturbations in a shared MR scenario. Future work should focus on collecting data from a larger and more diverse cohort of participants, balancing the occurrence of rare engagement levels, and studying how participant

Table 6: Attack-intensity ablation under the identical label-level stratified 4-fold protocol. Attack intensity alone is insufficient, while removing it degrades the model only modestly and still beats the traditional baselines that retain all features.

Feature regime	# Feat.	CV MAE
Full model	40	$\mathbf{0.89 \pm 0.19}$
Attack intensity only	1	1.07 ± 0.20
All except attack intensity	39	1.04 ± 0.11
Traditional baselines (all features)		
Ridge	40	1.09 ± 0.18
SVR	40	1.24 ± 0.15

Table 7: Verification results using *DeepPoly [28]* and *Prover [27]*. The verified sets correspond to parameter intervals where deterministic guarantees of engagement estimation hold.

System Parameter	Engagement ≥ 4	Engagement ≥ 5
Luminance	[110.2, 111.0]	[109.8, 110.6]
Optical Flow	[0.021, 0.024]	[0.019, 0.022]
Spatial Frequency	[943.5, 944.2]	[944.0, 945.2]
Temporal Smoothness	[7.1, 8.0]	—
Spectral Entropy	—	[7.01, 7.05]

background influences both reporting style and model behavior.

Measurement Fidelity. The model is trained entirely from self-report labels and does not yet incorporate objective engagement proxies such as task performance, physiological signals, or independent observer annotations. As a result, the learned target remains a subjective operationalization of engagement. Future studies should pair self-report with objective indicators so that the model can be validated against both perceived and externally measurable aspects of user state.

Advanced Modeling Architectures. While our two-phase LSTM-based approach proved effective, the baseline comparison shows that a traditional XGBoost regressor on temporally pooled features remains competitive on this dataset. Future research should therefore evaluate richer architectures only when they deliver clear gains in accuracy, interpretability, or deployment value. Attention-based architectures like Transformers may capture longer-range dependencies more effectively, and personalized fine-tuning may better account for individual differences in behavior and reporting styles.

Enhanced Verification Frameworks. Our initial application of formal verification provided sound, deterministic guarantees but resulted in narrow, conservative operating intervals. A key direction for future work is to develop a verification framework better suited to the stochastic nature of human cognitive states. This involves moving beyond deterministic guarantees toward probabilistic ones, which can provide wider, more practical operating ranges for system parameters while maintaining high confidence in the model's safety. Bridging this gap may involve integrating statistical methods with symbolic verification to produce guarantees that are both rigorous and directly applicable to real-world MR deployments. Building on such guarantees, a further direction is *closed-loop adaptation*, in which the estimated engagement and its derived safe-operating intervals drive runtime adjustment of controllable MR parameters to keep users engaged under cognitive attack.

8 Conclusion

We presented a two-phase modeling approach that combines contrastive pre-training with a supervised LSTM for predicting user engagement in Mixed Reality from multimodal time-series data under cognitive attack. In a controlled MR hostage-rescue study with step-wise latency perturbations, the model achieved an average MAE of 0.89 in label-level stratified cross-validation, outperforming Ridge, SVR, and

XGBoost baselines on temporally averaged features, and reached 0.84 MAE on fully held-out users. We also applied formal verification to derive conservative operating intervals for engagement-preserving MR parameter settings, establishing an initial bridge between engagement prediction and runtime reasoning.

Our work highlights several key directions for the future. The primary challenges are not only improving predictive accuracy, but also grounding engagement more fully in MR study design and making verification outputs more actionable for system designers. Future efforts will focus on expanding the dataset, incorporating objective engagement measures, evaluating when more complex models are genuinely necessary, and developing verification methods that provide wider and more practically useful guarantees for adaptive MR environments.

Acknowledgments

The human-subject data collection was approved by the Pennsylvania State University Institutional Review Board under Protocol STUDY00026827. The Northeastern University portion of the work involved only secondary analysis of de-identified data and was determined not to constitute human-subject research under IRB #25-09-06.

References

[1] I. B. Adhanom, P. MacNeilage, and E. Folmer. Eye tracking in virtual reality: a broad review of applications and challenges. *Virtual Reality*, 27(2):1481–1505, 2023. 1

[2] C. Austermann, F. Blanckenburg, K. Blanckenburg, and T. Utesch. Exploring the impact of virtual reality on presence: findings from a classroom experiment. In *Frontiers in Education*, vol. 10, p. 1560626. Frontiers Media SA, 2025. 3

[3] R. Chen, A. Andreyev, Y. Xiu, M. Imani, B. Li, M. Gorlatova et al. A neurosymbolic framework for interpretable cognitive attack detection in augmented reality. *arXiv preprint arXiv:2508.09185*, 2025. 2

[4] K. Cheng, J. F. Tian, T. Kohno, and F. Roesner. Exploring user reactions and mental models towards perceptual manipulation attacks in mixed reality. In *32nd USENIX Security Symposium (USENIX Security 23)*, pp. 911–928, 2023. 2

[5] B. David-John et al. Towards gaze-based prediction of the intent to interact in virtual reality. In *Proceedings of the 2021 Symposium on Eye Tracking Research and Applications (ETRA '21)*. ACM, 2021. doi: 10.1145/3448018.3458008 3

[6] P. B. Falola, A. E. Adeniyi, J. B. Awotunde, S. O. Akinola, M. Olagunju, and O. D. Olanloye. The role of augmented reality, virtual reality, and mixed reality in industries 4.0 and 5.0. In *Computational Intelligence in Industry 4.0 and 5.0 Applications*, pp. 329–356. Auerbach Publications, 2024. 1

[7] F. Garro, E. Fenoglio, M. Laffranchi, N. Garcia-Hernandez, and M. Semprini. Neurophysiological correlates of user engagement in mixed reality exergames. *Journal of Neural Engineering*, 22(3):036039, 2025. 1

[8] R. Gruen, E. Ofek, A. Steed, R. Gal, M. Sinclair, and M. Gonzalez-Franco. Measuring system visual latency through cognitive latency on video see-through ar devices. In *2020 IEEE conference on virtual reality and 3D user interfaces (VR)*, pp. 791–799. IEEE, 2020. 1

[9] K. L. Hobbs, M. L. Mote, M. C. Abate, S. D. Coogan, and E. M. Feron. Runtime assurance for safety-critical systems: An introduction to safety filtering approaches for complex control systems. *IEEE Control Systems Magazine*, 43(2):28–65, 2023. 2

[10] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 3

[11] J. Hu and W. Xiao. What are the influencing factors of online learning engagement? a systematic literature review. *Frontiers in Psychology*, 16:1542652, 2025. 3

[12] S. Irshad and A. Perkis. Increasing user engagement in virtual reality:the role of interactive digital narratives to trigger emotional responses. In *Proceedings of the 11th Nordic Conference on Human-Computer Interaction: Shaping Experiences, Shaping Society*, NordiCHI '20, art. no. 106, 4 pp. Association for Computing Machinery, New York, NY, USA, 2020. doi: 10.1145/3419249.3421246 2

[13] R. Islam, Y. Lee, M. Jaloli, I. Muhammad, D. Zhu, P. Rad et al. Automatic detection and prediction of cybersickness severity using deep neural networks from user's physiological signals. In *2020 IEEE international symposium on mixed and augmented reality (ISMAR)*, pp. 400–411. IEEE, 2020. 3

[14] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P. A. Muller. Deep learning for time series classification: a review. *Data Mining and Knowledge Discovery*, 33(4):917–963, 2019. 3

[15] A. T. Jebb, L. Tay, W. Wang, and Q. Huang. Time series analysis for psychological research: examining and forecasting change. *Frontiers in Psychology*, 6:727, 2015. 3

[16] A. Joshi, S. Kale, S. Chandel, and D. K. Pal. Likert scale: Explored and explained. *British journal of applied science & technology*, 7(4):396, 2015. 2

[17] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola et al. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673, 2020. 2, 3

[18] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[19] J. Li and P. O. Kristensson. On the benefits of sensorimotor regularities as design constraints for superpower interactions in mixed reality. *IEEE Transactions on Visualization and Computer Graphics*, 2025. 1

[20] S. Li, C. Gao, J. Zhang, Y. Zhang, Y. Liu, J. Gu et al. Less cybersickness, please: Demystifying and detecting stereoscopic visual inconsistencies in virtual reality apps. *Proceedings of the ACM on Software Engineering*, 1(FSE):2167–2189, 2024. 3

[21] Z. Li, Y. Ji, R. Chen, T. Liu, Y. Wang, Y. Shi et al. Modeling the impact of visual stimuli on redirection noticeability with gaze behavior in virtual reality. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, CHI '25, art. no. 38, 18 pp. Association for Computing Machinery, New York, NY, USA, 2025. doi: 10.1145/3706598.3713392 2

[22] Y. Liu, L. Zhao, and C. Zhang. Self-supervised contrastive learning for time-series: A systematic review. *ACM Computing Surveys*, 56(2):1–35, 2023. 3

[23] I. S. MacKenzie and C. Ware. Lag as a determinant of human performance in interactive systems. In *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems*, pp. 488–493. ACM, 1993. doi: 10.1145/169059.169431 2

[24] Z. Merchant, E. T. Goetz, L. Cifuentes, W. Keeney-Kennicutt, and T. J. Davis. Effectiveness of virtual reality-based instruction on students' learning outcomes in k-12 and higher education: A meta-analysis. *Computers & Education*, 70:29–40, 2014. 2

[25] A. Pavlovych and W. Stuerzlinger. Target following performance in the presence of latency, jitter, and signal dropouts. In *Proceedings of Graphics Interface 2011 (GI '11)*, pp. 33–40, 2011. 2

[26] J. Radianti, T. A. Majchrzak, J. Fromm, and I. Wohlgenannt. A systematic review of immersive virtual reality applications for higher education: Design elements, lessons learned, and research agenda. *Computers & Education*, 147:103778, 2020. 2

[27] W. Ryou, J. Chen, M. Balunovic, G. Singh, A. Dan, and M. Vechev. Scalable polyhedral verification of recurrent neural networks. In *Computer Aided Verification: 33rd International Conference, CAV 2021*, pp. 225–248. Springer-Verlag, Berlin, Heidelberg, 2021. 3, 5, 9

[28] G. Singh, T. Gehr, M. Püschel, and M. Vechev. An abstract domain for certifying neural networks. *Proc. ACM Program. Lang.*, 3(POPL), art. no. 41, 30 pp., Jan. 2019. 3, 5, 9

[29] J. T. Slagel, L. M. White, A. Dutle, C. A. Muñoz, and N. Crespo. A formal verification framework for runtime assurance. In N. Benz, D. Gopinath, and N. Shi, eds., *NASA Formal Methods*, pp. 322–328. Springer Nature Switzerland, Cham, 2024. 2

[30] R. Somarathna, D. S. Elvitigala, Y. Yan, A. J. Quigley, and G. Mohammadi. Exploring user engagement in immersive virtual reality games through multimodal body movements. In *Proceedings of the 29th ACM Symposium on Virtual Reality Software and Technology*, pp. 1–8, 2023. 2

[31] T. Thoyyibah, W. Haryono, A. U. Zailani, Y. M. Djaksana, N. Rosmawarni, and N. D. Arianti. Transformers in machine learning: literature review. *Jurnal Penelitian Pendidikan IPA*, 9(9):604–610, 2023. 3

[32] P. Viitaharju, M. Nieminen, J. Linnera, K. Yliniemi, and A. J. Karttunen. Student experiences from virtual reality-based chemistry laboratory exercises. *Education for Chemical Engineers*, 44:191–199, 2023. 3

[33] T. Waltemate, I. Senna, F. Hülsmann, M. Rohde, S. Kopp, M. Ernst et al. The impact of latency on perceptual judgments and motor performance in closed-loop interaction in virtual reality. In *Proceedings of the 22nd ACM Conference on Virtual Reality Software and Technology (VRST '16)*, pp. 27–35. ACM, 2016. doi: 10.1145/2993369.2993381 2

[34] D. Wang et al. Assessing virtual reality presence through physiological

measures: A review. *Frontiers in Virtual Reality*, 2025. 3

[35] T.-W. Weng, H. Zhang, H. Chen, J. Yi, D. Su, Y. Gao et al. Towards fast computation of certified robustness for relu networks. In *International Conference on Machine Learning (ICML)*, pp. 5273–5282, 2018. 3

[36] P. Wu, N. Ahmed, K. Huang, T. Lan, and M. Imani. Personalized bayesian networks for cybersickness prediction in virtual reality. In *Proceedings of the 1st Workshop on Enhancing Security, Privacy, and Trust in Extended Reality (XR) Systems (XR Security 2025)*. Houston, TX, USA, Sept. 2025. 2

[37] P. Wu, N. Ahmed, A. Sarma, K. Huang, R. Islam, B. Li et al. Probabilistic verification of cybersickness in virtual reality through bayesian networks. In *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2025. 3

[38] T. L. Xu, K. De Barbaro, D. H. Abney, and R. F. Cox. Finding structure in time: Visualizing and analyzing behavioral time series. *Frontiers in Psychology*, 11:1457, 2020. 3

[39] C. Zhang, Q. Yan, L. Meng, and T. Sylvain. What constitutes good contrastive learning in time-series forecasting? *arXiv preprint arXiv:2306.12086*, 2023. 3