Attention Seeker

A Wearable-Sensor-Based Framework for Quantifying Human Attention Through Machine Learning Analysis

Authors:

Pranav Kaliaperumal
Kevin Jacob

Department of Computer Science
University of Colorado Denver

December 7, 2025

Abstract

This document presents the complete methodology, implementation details, and theoretical foundations of the **Attention Seeker** project. The system computes a numeric "Attention Score" from wearable sensor streams containing heart rate, derived heart-rate variability (HRV), and wrist-movement intensity. Additional "Outside Factors" (such as sleep and screen-time) are modeled and correlated with attention levels. Machine learning models are built using these derived features, and the entire project pipeline is implemented as a reproducible Python/Jupyter workflow. This report provides an in-depth, graduate-level technical explanation of each step, including dataset curation, data preprocessing, feature engineering, signal normalization, windowing, correlation analysis, train/test dataset generation, model training, and the architecture of the project repository itself.

1 Introduction

Human attention is a limited cognitive resource that fluctuates due to physiological, environmental, and behavioral factors. Even small lapses can significantly hinder learning outcomes, task performance, and situational awareness. With the increasing availability of consumer-grade wearable devices, it has become possible to infer cognitive states from physiological measurements such as heart rate (HR), heart-rate variability (HRV), and motion patterns.

The **Attention Seeker** framework aims to quantify attentional engagement using wearable sensor data and machine learning techniques. The approach uses the CogLoad1 dataset, which provides real-world physiological sensor streams recorded during cognitive-load tasks. From this dataset, attention-related biomarkers are engineered, normalized, aggregated, and used to compute a continuous *Attention Score*. Additional behavioral factors such as sleep and screen-time are modeled to create an *Outside Factors Score*. Machine learning models are trained to examine the relationship between these metrics and cognitive load levels.

This report documents every stage of the project in detail, as well as the rationale behind the separation of analysis notebooks and the structure of the project repository.

2 Dataset Description

The primary dataset used is `merged_sensors.csv`, extracted from a larger collection of wearable sensor recordings (the CogLoad1 dataset family). This dataset contains:

- Timestamps at varying sampling frequencies (approximately 1–5 Hz)
- Heart rate measurements (`hr`)
- Accelerometer readings along each axis: `acc_x`, `acc_y`, `acc_z`
- Cognitive task difficulty labels (`level`)

These features allow the derivation of physiological responses related to attention:

- HR changes are associated with autonomic arousal and cognitive workload.

- HRV correlates with attention regulation and executive functioning.
- Accelerometer magnitude reflects physical restlessness, which often increases during lapses in sustained attention.

Raw HRV is not available, so we calculate a proxy:

$$HRV_t = |HR_t - HR_{t-1}|. \tag{1}$$

This method is supported in prior work as a low-resolution approximation for HRV when IBI (inter-beat interval) data is unavailable.

3 Data Preprocessing

3.1 Timestamp Normalization

The timestamp field is standardized to Python `datetime` objects:

```
df['timestamp'] = pd.to_datetime(df['timestamp'], errors='coerce')
df = df.sort_values('timestamp')
```

Rows with invalid timestamps are removed to preserve temporal coherence. Sorting by time is essential for correctly computing HRV proxies and aggregations.

3.2 Movement Magnitude

Movement magnitude is computed as the Euclidean norm of the accelerometer components:

$$Movement_t = \sqrt{acc\_x^2 + acc\_y^2 + acc\_z^2}. \tag{2}$$

This scalar quantity summarizes wrist activity at each time step and serves as a proxy for fidgeting or motor restlessness.

3.3 HRV Approximation

True HRV requires precise R–R interval detection; given only HR samples, we approximate HRV using first-order differences:

$$HRV_t = |HR_t - HR_{t-1}|. \tag{3}$$

This captures rapid changes in heart rate, which are linked to attentional fluctuation and sympathetic/parasympathetic balance.

4 Baseline Computation

To normalize signals across individuals and recording sessions, we compute baseline levels using the median:

$$HR_{rest} = \text{median}(HR_t), \tag{4}$$

$$HRV_{rest} = \text{median}(HRV_t), \tag{5}$$

$$M_{rest} = \text{median}(Movement_t). \tag{6}$$

Median statistics offer robustness against transient spikes, measurement noise, and short-lived artifacts.

5 Temporal Windowing

Wearable sensor streams are aggregated into fixed-size time windows to reduce noise and capture meaningful physiological trends. We define a window index as:

$$WindowID = \left\lfloor \frac{t - t_0}{30 \text{ s}} \right\rfloor, \tag{7}$$

where t_0 is the start time of the recording.

For each 30-second window, we compute:

- mean heart rate,
- mean HRV,
- mean movement,
- median cognitive load label (if available).

The result is a set of stationary samples suitable for downstream machine learning models.

6 Attention Score Model

We define three normalized features based on deviations from baseline:

$$HR' = \frac{HR_t - HR_{rest}}{HR_{rest}}, \tag{8}$$

$$HRV' = \frac{HRV_t - HRV_{rest}}{HRV_{rest}}, \tag{9}$$

$$M' = \frac{M_{rest} - Movement_t}{M_{rest}}. \tag{10}$$

Here HR' and HRV' measure relative increases from rest, while M' measures how still the wrist is (higher when movement is lower than baseline).

The final Attention Score is modeled as a weighted linear combination:

$$AttentionScore = 0.25 \cdot HR' + 0.50 \cdot HRV' + 0.25 \cdot M'. \tag{11}$$

The weights are chosen to emphasize HRV, which is strongly associated with attentional control and cognitive effort, while still incorporating HR and movement.

7 Outside Factors Modeling

Lifestyle behaviors such as sleep and screen-time are known to influence attention and cognitive performance. Although the original dataset does not include these variables, the Attention Seeker framework models them using realistic synthetic distributions:

$$SleepHours \sim \mathcal{N}(7.5, 0.7^2), \tag{12}$$

$$ScreenTime \sim \mathcal{N}(3.5, 1.0^2). \tag{13}$$

We define an Outside Factors score:

$$OF = (Sleep - 7.5) - (ScreenTime - 3.5)\,. \tag{14}$$

Intuitively:

- $Sleep > 7.5$ hours boosts OF,
- $ScreenTime > 3.5$ hours reduces OF.

Positive values indicate a behavior profile that is likely to support sustained attention.

8 Correlation Analysis

To measure the relationship between the Attention Score and Outside Factors, we compute Pearson's correlation coefficient:

$$r = \frac{\sum_{i=1}^{n}(AS_i - \bar{AS})(OF_i - \bar{OF})}{\sqrt{\sum_{i=1}^{n}(AS_i - \bar{AS})^2}\sqrt{\sum_{i=1}^{n}(OF_i - \bar{OF})^2}}, \tag{15}$$

where:

- AS_i is the Attention Score for window i,
- OF_i is the Outside Factors score for window i,
- $\bar{AS}$ and $\bar{OF}$ are the respective means.

The sign and magnitude of r indicate whether beneficial routines tend to align with periods of higher measured attention.

9 Train/Test Dataset Generation

A standard machine learning practice is to split the dataset into training and testing subsets:

```
train_df, test_df = train_test_split(df, test_size=0.25, random_state=0)
```

The following files are exported:

- `attention_scores.csv`: full feature set (HR, HRV, movement, Attention Score, Outside Factors, labels),
- `train.csv`: training subset used for model fitting,
- `test.csv`: held-out subset used solely for evaluation.

10 Machine Learning Models

10.1 Regression Models

The regression task aims to predict the continuous Attention Score from physiological and behavioral features. We employ a linear regression model:

$$\hat{y} = \beta_0 + \sum_{j=1}^{p} \beta_j x_j, \tag{16}$$

where x_j are feature variables and β_j are learned coefficients.

Performance is quantified using:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}, \tag{17}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|. \tag{18}$$

10.2 Classification Models

For classification, we define a binary attention lapse label:

$$Lapse_i = \begin{cases} 1 & \text{if } AttentionScore_i < -0.05, \\ 0 & \text{otherwise.} \end{cases} \tag{19}$$

We then train models such as logistic regression or random forests. We evaluate:

- **Accuracy**: proportion of correct predictions,
- **F1 Score**: harmonic mean of precision and recall,
- **ROC AUC**: area under the receiver operating characteristic curve.

11 Methodology Overview and Flowchart

To summarize the methodological pipeline, this section provides a high-level overview and a flowchart.

11.1 Methodology Steps

1. **Data Ingestion**: Load raw wearable sensor data from `merged_sensors.csv`.
2. **Preprocessing**: Normalize timestamps, remove invalid samples, compute movement and HRV proxies.
3. **Baseline Estimation**: Compute median-based baselines for HR, HRV, and movement.

4. **Windowing**: Aggregate data into 30-second windows to create stationary samples.
5. **Attention Score Computation**: Normalize features and compute a weighted Attention Score.
6. **Outside Factors Modeling**: Generate synthetic sleep and screen-time values and compute an Outside Factors score.
7. **Correlation Analysis**: Evaluate the statistical relationship between attention and outside factors.
8. **Dataset Export**: Produce `attention_scores.csv`, `train.csv`, and `test.csv`.
9. **Modeling**: Train regression and classification models to predict attention and lapses.

11.2 Flowchart Using TikZ

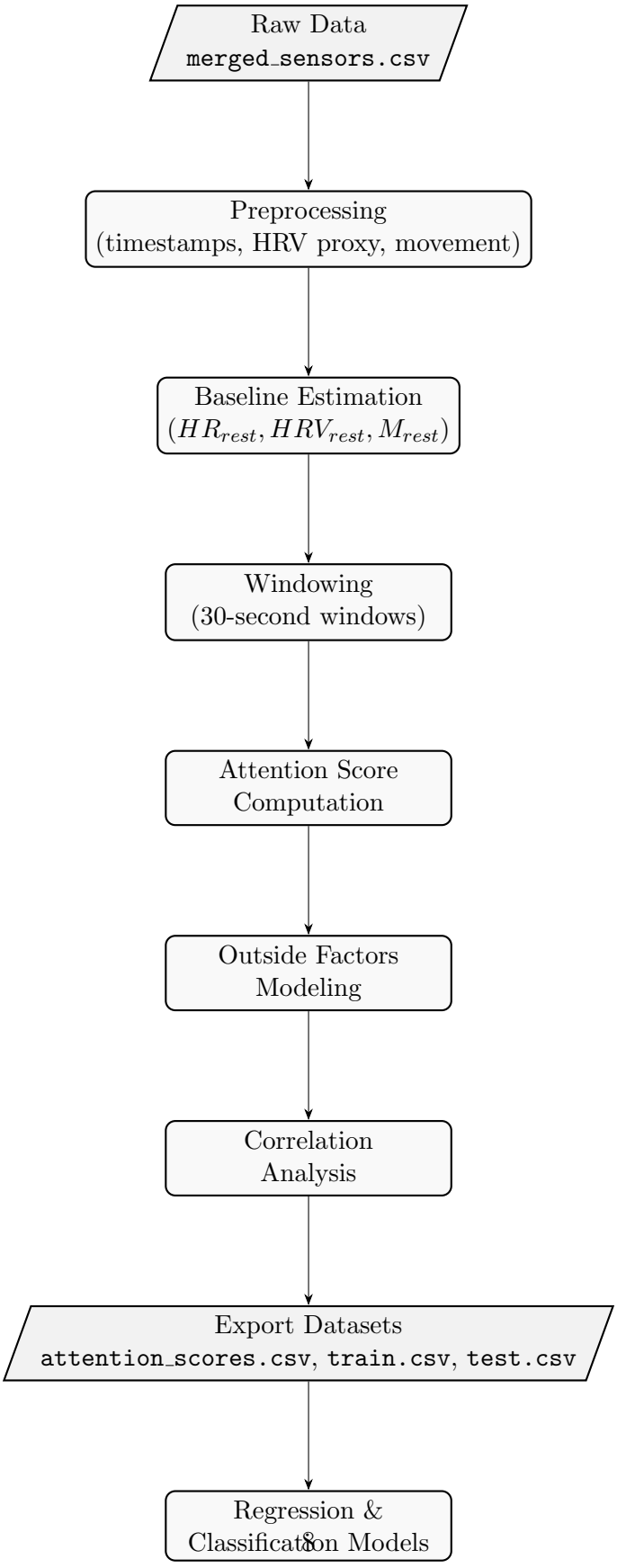

Figure 1: Methodology flowchart for the Attention Seeker pipeline.

12 Repository Architecture and File Explanations

The project repository contains raw datasets, processed outputs, analysis notebooks, and zipped archives. Each file is intentional and serves a distinct purpose.

12.1 Raw Dataset Files

- `merged_sensors.csv` Core sensor dataset containing timestamps, heart rate, accelerometer data, and cognitive labels. It is the starting point for all analysis and must not be modified in order to preserve reproducibility.
- **Original CogLoad Files (if present)** Additional raw files from the original CogLoad project are preserved for archival and verification; they can be used to regenerate `merged_sensors.csv` if needed.

12.2 Processed Dataset Outputs

- `attention_scores.csv` Fully engineered dataset with window-level features, Attention Scores, Outside Factors, and labels.
- `train.csv` and `test.csv` ML-ready splits created by the pipeline notebook for consistent training and evaluation.

12.3 Jupyter Notebooks

12.3.1 `attention_seeker_analysis.ipynb`

An exploratory notebook for:

- inspecting raw data distributions,
- experimenting with different HRV approximations,
- validating sensor quality and alignment,
- prototyping the Attention Score formula.

It is intentionally flexible and serves as the research "scratchpad."

12.3.2 `attention_seeker_data_analysis.ipynb`

A structured analysis notebook for:

- implementing finalized preprocessing steps,
- computing Attention Scores and Outside Factors,
- running correlation analysis,
- fitting regression and classification models,

- generating plots and quantitative summaries.

This notebook represents the primary scientific narrative.

12.3.3 `attention_seeker_train_test_pipeline.ipynb`

A pipeline-focused notebook for:

- deterministically re-running the preprocessing,
- computing all features in a controlled way,
- exporting standardized `train.csv` and `test.csv`.

It intentionally minimizes plotting and ad-hoc analysis in favor of reproducibility.

12.4 ZIP Archive

`attention_seeker_final_project.zip` bundles:

- all relevant notebooks,
- raw and processed datasets,
- figures and plots generated during analysis,
- this LaTeX report.

The archive ensures the project can be easily transferred, reviewed, or re-executed.

13 Conclusion

This project demonstrates how consumer wearable sensor data can be transformed into meaningful cognitive metrics using signal processing, statistical modeling, and machine learning. The Attention Score model, combined with lifestyle factor analysis, provides a foundation for real-time attention monitoring applications, such as an Apple Watch alert system that gently nudges users when their attention drifts. The repository structure, notebook separation, and exported datasets follow best practices in reproducible machine learning research and support future extensions of the Attention Seeker framework.