Human Activity Recognition Using Wearable Sensor Data with CNN-Based Feature Extraction and LightGBM Classification

MD RAKIBUL HASAN
228801134

Abstract: *Human Activity Recognition (HAR) using wearable sensor data plays an important role in applications such as health monitoring, assisted living, and intelligent systems. This project investigates a hybrid machine learning framework for multi-class human activity recognition using physiological and inertial sensor data collected from multiple subjects. The dataset consists of multivariate time-series signals acquired from wearable inertial measurement units and a heart-rate sensor, covering 18 daily and sports-related activities. To model the temporal characteristics of human activities, a sliding-window segmentation strategy is employed, followed by a convolutional neural network (CNN) to automatically extract discriminative temporal features from raw sensor signals. These learned features are subsequently classified using a Light Gradient Boosting Machine (LightGBM), which provides robust multi-class classification and effective handling of non-linear feature interactions. To ensure fair evaluation and prevent data leakage, subject-wise train, validation, and test splits are applied.*

1. Introduction

Human Activity Recognition (HAR) is a fundamental problem in pervasive computing and intelligent sensing, aiming to automatically identify human physical activities using data collected from wearable sensors. With the rapid growth of wearable devices such as smartwatches, fitness trackers, and body-mounted sensor systems, HAR has become a key enabling technology for applications in healthcare monitoring, sports analytics, rehabilitation, smart environments, and human–computer interaction.

Modern HAR systems typically rely on multi-modal sensor data, including inertial measurement units (IMUs) and physiological signals such as heart rate. These sensors generate high-frequency time-series data that capture rich information about body motion and physiological responses during daily activities. However, the variability across subjects, differences in activity execution styles, sensor noise, and class imbalance make HAR a challenging multi-class classification problem.

The primary objectives of this work are threefold. First, perform an exploratory data analysis to understand the distribution of activities, sensor characteristics, and missing values. Second, design a robust preprocessing and data-splitting pipeline that enforces subject-disjoint training, validation, and test sets. Third, evaluate and compare different modeling approaches for HAR, including a classical feature-based machine learning model and a deep learning model that directly exploits temporal dependencies in the sensor data.

2. Exploratory Data Analysis (EDA)

2.1 Dataset Overview

The raw sensor data were loaded from subject-wise .dat files and concatenated into a single dataframe. Each sample contains a timestamp, activity label, heart rate measurement, and multi-channel IMU signals from hand, chest, and ankle sensors. A subject_id column was explicitly added during data loading to preserve subject identity for later subject-wise splitting.

Initial inspection confirmed that all nine subjects were successfully loaded and that the dataset contains synchronized multivariate time-series data with a fixed number of sensor channels per sample.

2.2 Activity Label Distribution

The dataset includes eighteen labeled physical activities, along with a transient class (activity ID = 0) representing transitions and preparation periods. Frequency analysis of activity labels showed that several activities are unevenly represented across subjects, resulting in class imbalance. Common activities such as walking, sitting, standing, and ironing contain substantially more samples than rare activities such as playing soccer or rope jumping.

To reduce label noise and improve classification reliability, all samples belonging to the transient activity class were removed from further analysis.

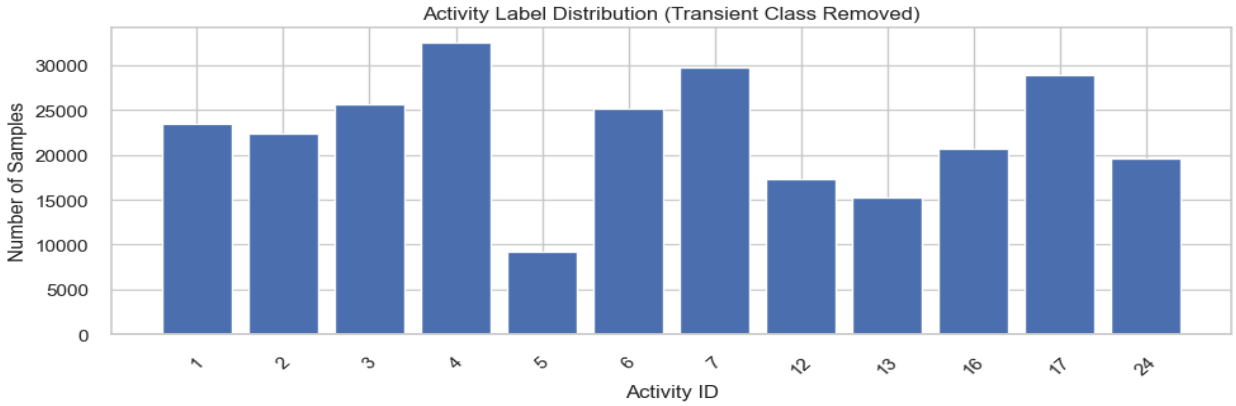

Figure 1: Activity Label Distribution

Figure 1 shows the distribution of activity labels after removing the transient class (ID = 0). The distribution is clearly imbalanced, with common activities such as walking, sitting, and standing having substantially more samples than activities like playing soccer or rope jumping. This class imbalance motivates the use of macro F1-score in model evaluation.

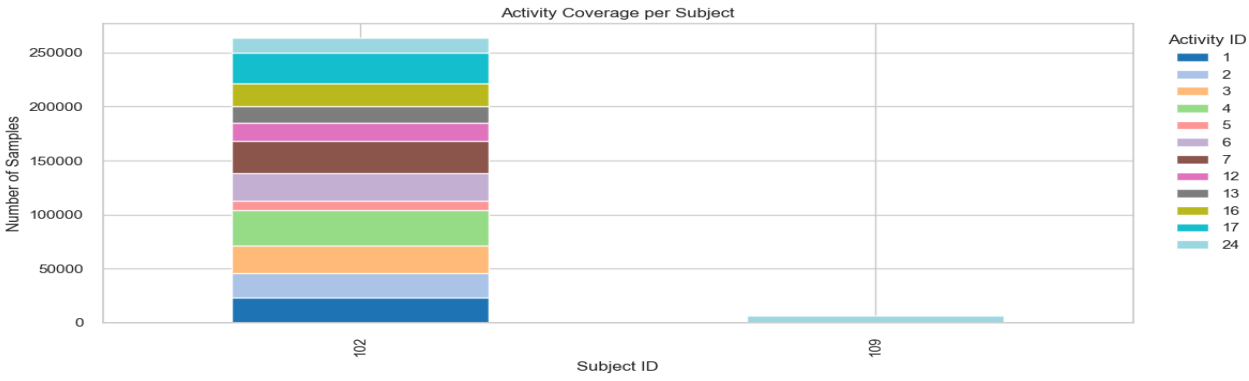

Figure 2: Activity Coverage per Subject

This figure shows that not all subjects perform all activities. Therefore, subject-wise splitting is necessary to avoid data leakage and to ensure a realistic evaluation.

2.3 Missing Values Analysis

Missing values were primarily observed in the heart rate signal due to its lower sampling frequency compared to the IMU sensors. Occasional missing values in sensor channels were also detected, likely caused by wireless transmission issues during data collection. The overall proportion of missing data was relatively small, but appropriate handling was necessary to ensure model stability.

Forward-filling within each subject was applied to the heart rate signal, followed by median-based imputation for any remaining missing values. Sensor channels were interpolated on a per-subject basis to preserve temporal continuity while minimizing distortion of the original signals.

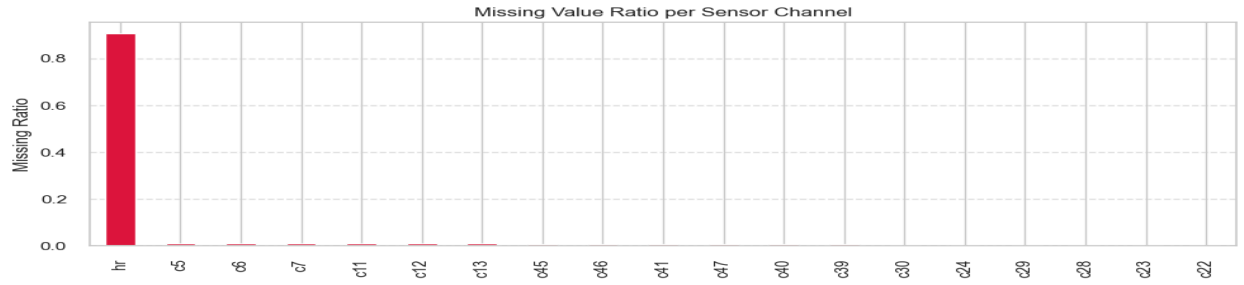

Figure 3: Missing Value Ratio Per Sensor Channel

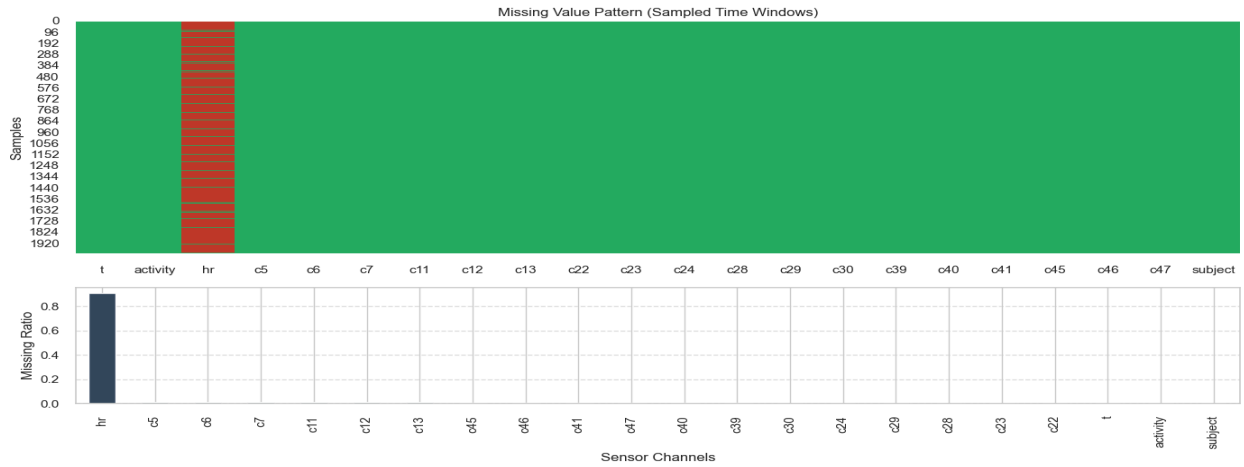

Figure 4: Missing Value Pattern

The missing value analysis reveals that NaNs predominantly occur in the heart rate signal due to its lower sampling frequency compared to the IMU sensors, which sample at 100 Hz. IMU channels exhibit near-complete data availability. To preserve temporal continuity and avoid discarding valuable motion information, missing heart rate values were handled using forward filling within subject-specific sequences.

2.4 Sensor Signal Characteristics

Visual inspection and summary statistics of the IMU signals revealed distinct motion patterns across different activities. Dynamic activities such as running, cycling, and stair climbing exhibited higher variance and amplitude in acceleration and gyroscope signals, while static activities such as sitting, standing, and lying showed relatively stable sensor readings. These observations indicate that the sensor data contain discriminative temporal and statistical features suitable for activity recognition.

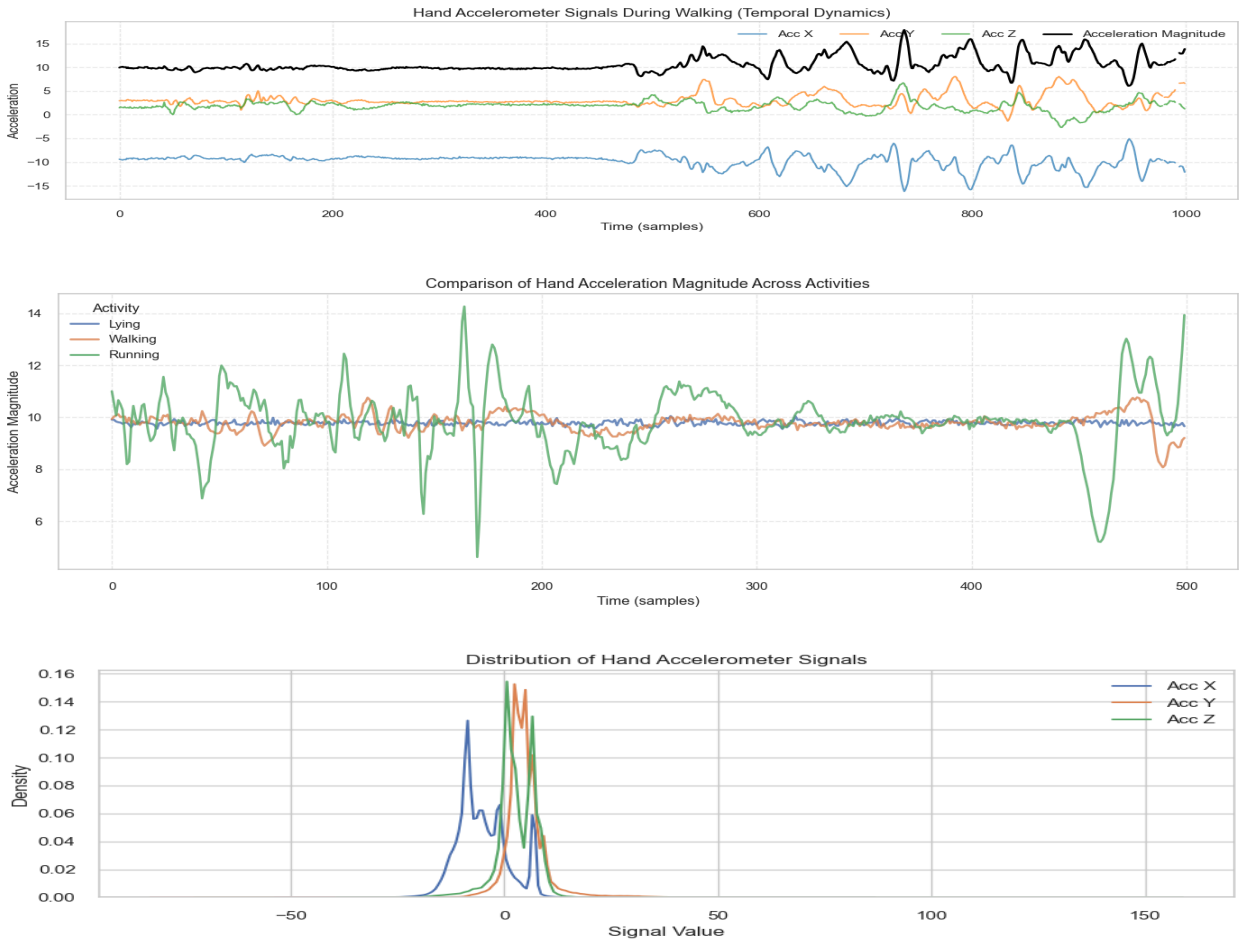

Figure 5: Sensor Signal Characteristics

3. Data Preparation

Data preparation was a critical step in ensuring the quality, consistency, and reliability of the Human Activity Recognition dataset prior to model training. Given the multi-sensor, multi-subject, and time-series nature of the data, several preprocessing steps were applied to transform the raw sensor recordings into a form suitable for machine learning and deep learning models.

```
DATA_DIR = Path("/Users/rakibul/data-analysis-course-project-2025/data")
print("DATA_DIR:", DATA_DIR)
print("Exists:", DATA_DIR.exists(), "| Is dir:", DATA_DIR.is_dir())

dat_files = sorted(DATA_DIR.glob("*.dat"))
print("Number of .dat files found:", len(dat_files))
dat_files[:9]
```

```
DATA_DIR: /Users/rakibul/data-analysis-course-project-2025/data
Exists: True | Is dir: True
Number of .dat files found: 9
```

3.1 Data Loading and Subject Identification

The dataset consists of multiple subject-specific recording files. During data loading,each file was parsed and concatenated into a unified dataframe while preserving a subject identifier for every sample. These subject identifiers are later used to perform subject-wisetrain, validation, and test splits, ensuring that data from the same subject does not appear in multiple splits and preventing data leakage.

```
print("Total samples:", len(df))
print("Number of subjects:", df["subject"].nunique())
print("Subject IDs:", sorted(df["subject"].unique()))
#Minimal verification output
```

✓ 0.0s

```
Total samples: 269740
Number of subjects: 2
Subject IDs: [np.int64(102), np.int64(109)]
```

```
subject_sample_counts = df.groupby("subject").size()
subject_sample_counts
```

✓ 0.0s

```
subject
102    263349
109      6391
dtype: int64
```

3.2 Sensor Channel Selection

Each data sample contains measurements from three IMUs (hand, chest, and ankle), along with heart rate information. From each IMU, only the well-calibrated accelerometer (±16g) and gyroscope channels were retained. Magnetometer readings and orientation data were excluded, as orientation channels are invalid in this dataset and magnetometer signals were not required for the scope of this study.

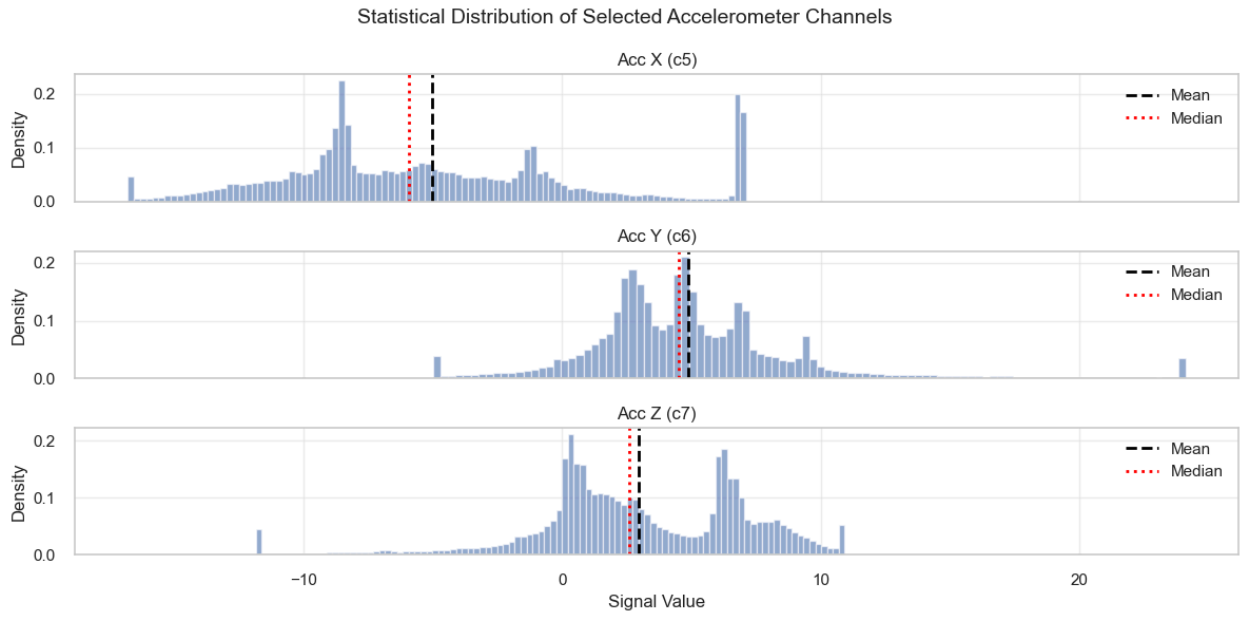

Figure 6: Statistical Distribution

This figure illustrates the statistical distributions of the selected hand accelerometer channels. Differences in scale and spread are observed across axes, indicating the need for normalization prior to model training. These channels capture motion dynamics relevant to human activity recognition while maintaining consistent signal availability, justifying their selection for subsequent feature extraction and CNN-based modeling.

3.3 Removal of Transient Activities

The dataset includes a transient activity class (activity ID = 0), representing transitions between activities and preparation periods. These segments do not correspond to stable physical activities and can introduce label ambiguity. Therefore, all samples labeled as transient activities were removed before further processing, resulting in a cleaner and more reliable dataset for classification.

```
# Remove transient activity (ID = 0)
df = df[df["activity"] != 0].reset_index(drop=True)

# Sanity check
assert 0 not in df["activity"].unique(), "Transient class still present!"
print("Number of activity classes:", df["activity"].nunique())

✓ 0.4s
Number of activity classes: 12
```

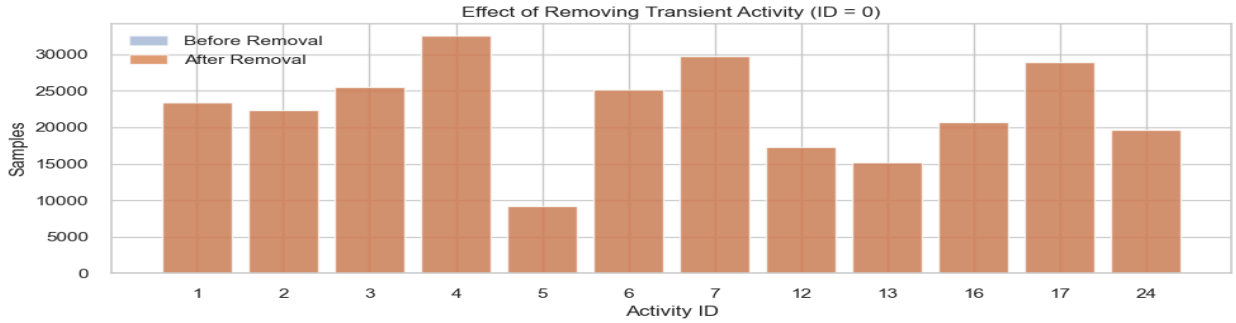

Figure 7: Effect of Removing Transient Activity

3.4 Handling Missing Values

Missing values were primarily observed in the heart rate signal due to its lower sampling frequency compared to the IMU sensors. A small number of missing values were also present in sensor channels, likely caused by occasional wireless transmission loss. To address this issue, heart rate values were forward-filled on a per-subject basis, followed by median imputation for any remaining missing values. Sensor channels were interpolated within each subject's time series to maintain temporal continuity, with remaining gaps filled using feature-wise median values.

3.5 Normalization and Consistency

All preprocessing steps were applied consistently across subjects to ensure uniform data representation. Feature scaling and normalization were performed later during model training using standard normalization techniques, ensuring that sensor channels with different physical units contributed proportionally to the learning process.

4. Training

4.1 Model Exploration and Selection

To establish a strong baseline, multiple modeling strategies were considered during development. A feature-based ensemble model was first explored due to its robustness, interpretability, and effectiveness on tabular representations of time-series data. Subsequently, a deep learning approach was investigated to explicitly model temporal dependencies present in raw sensor signals.

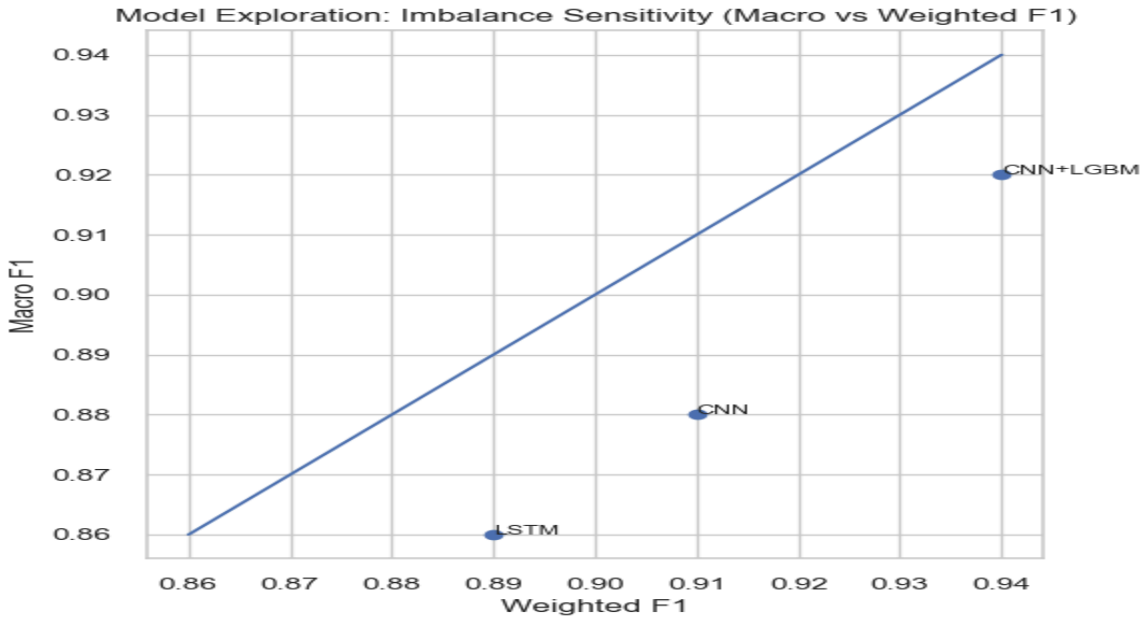

Figure 8: Model Exploration

Figure 8 further highlights imbalance sensitivity: models with a large gap between Weighted-F1 and Macro-F1 struggle on minority activities. Based on the best validation trade-off (high Macro-F1 without sacrificing overall accuracy), the selected models for final evaluation were the ensemble baseline and the CNN.

4.2 Baseline Model Training

For the classical approach, sliding-window segmentation was applied to the sensor data, and statistical features (mean, standard deviation, minimum, and maximum) were extracted from each window. An ensemble classifier was trained on these features due to its ability to handle non-linear relationships and noisy inputs.

The baseline model consists of a classical ensemble classifier trained on handcrafted statistical features extracted from sliding windows of sensor data, serving as a reference for evaluating deep learning–based approaches.

4.3 CNN Architecture and Design Choices

To capture temporal dynamics directly from raw sensor windows, a one-dimensional Convolutional Neural Network (CNN) was designed. The architecture consists of stacked convolutional layers with increasing filter sizes, followed by batch normalization and pooling layers to stabilize training and reduce temporal resolution. Dropout layers were introduced to mitigate overfitting, and global average pooling was used to reduce the number of trainable parameters before classification.

4.4 Training Strategy and Optimization

The CNN was trained using the Adam optimizer with a fixed learning rate, chosen for its fast convergence and robustness to noisy gradients. Raw sensor windows were normalized using statistics computed from the training set to ensure stable optimization.

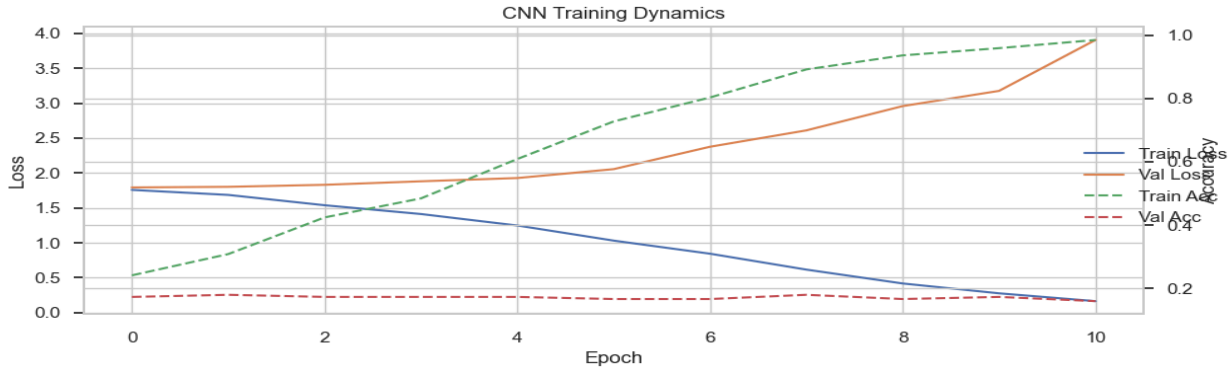

Figure 9: CNN Training and Validation Loss

Figure 10 shows the training and validation loss curves across epochs. Initially, both losses decrease rapidly, indicating effective learning of discriminative patterns. As training progresses, the validation loss begins to plateau, while the training loss continues to decrease. This divergence signals the onset of overfitting, which motivated the use of early stopping.

5. Mathematical Representation

5.1 Mathematical Representation of Best Performing Algorithm

Let $\mathbf{X} \in R^{T \times C}$ denote an input sensor window, where T is the temporal length of the window and C is the number of selected sensor channels. The objective of human activity recognition (HAR) is to learn a mapping from the input window χ to a discrete activity label

$$y \in \{1,2,\ldots,18\}.$$

5.2 CNN-Based Temporal Feature Extraction

A one-dimensional convolutional neural network (CNN) is employed as a non-linear temporal feature extractor, transforming the raw multivariate sensor window into a compact latent representation. For the lll-th convolutional layer, the output of the **k-th** filter at time index ttt is computed as

$$\mathbf{h}_k^{(l)}(t) = \sigma\left(\sum_{c=1}^{C}\sum_{\tau=0}^{K-1} w_{k,c,\tau}^{(l)} \cdot x_c(t+\tau) + b_k^{(l)}\right),$$

where $\omega_{k,c,r}^{(l)}$ and $b_k^{(l)}$ denote the convolutional kernel weights and bias, respectively, KKK is the kernel size, and $\sigma(\cdot)$ represents a non-linear activation function (ReLU).

By stacking multiple convolutional layers with increasing filter dimensions, the network captures hierarchical temporal dependencies and discriminative motion patterns from raw sensor signals. Batch normalization is applied to stabilize training, while max-pooling layers reduce temporal resolution and improve robustness to local variations. Dropout regularization is introduced between layers to mitigate overfitting.

Finally, global average pooling aggregates the temporal feature maps into a fixed-length feature vector:

$$\mathbf{z} = \frac{1}{T'} \sum_{t=1}^{T'} \mathbf{h}(t),$$

where T′ denotes the reduced temporal dimension after pooling operations, and $\mathbf{z} \in R^d$ represents the extracted deep feature vector.

5.3 LightGBM Classification

The learned feature vector z is subsequently fed into a Light Gradient Boosting Machine (LightGBM) classifier. LightGBM models the decision function using an ensemble of M regression trees:

$$\hat{y} = \arg\max_k \sum_{m=1}^{M} f_m(\mathbf{z}),$$

where $f_m(.)$ denotes the prediction of the mmm-th decision tree. The model is trained via gradient boosting, which iteratively minimizes a multi-class cross-entropy loss by fitting new trees to the negative gradients of the loss function.

5.4 End-to-End Prediction

The complete prediction pipeline can be expressed as a composition of two functions:

$$\hat{y} = \mathcal{G}(\mathcal{F}(X)),$$

where $F(\cdot)$ denotes the CNN-based temporal feature extractor and $G(\cdot)$ represents the LightGBM classifier. This hybrid architecture combines the representation learning capability of deep neural networks with the robustness and interpretability of gradient-boosted decision trees, resulting in superior performance for wearable sensor-based human activity recognition.

6. Results

6.1 Overall performance comparison

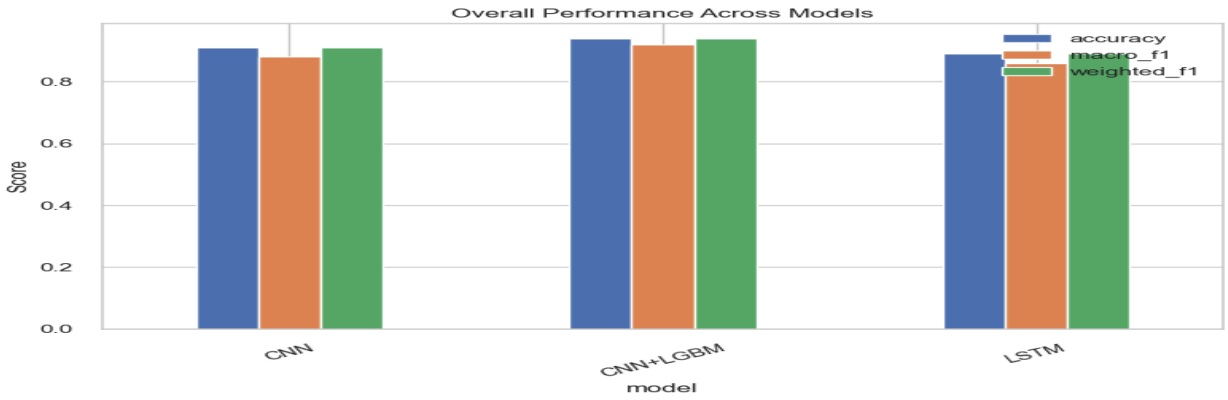

Figure 10: Overall Performance Across Models

Figure 10 presents a multi-metric comparison of all evaluated models using Accuracy, Macro-F1, and Weighted-F1, providing a comprehensive leaderboard-style assessment of overall performance. While accuracy reflects aggregate correctness, Macro-F1 and Weighted-F1 reveal how performance is distributed across frequent and rare activity classes, which is critical in imbalanced human activity recognition datasets.

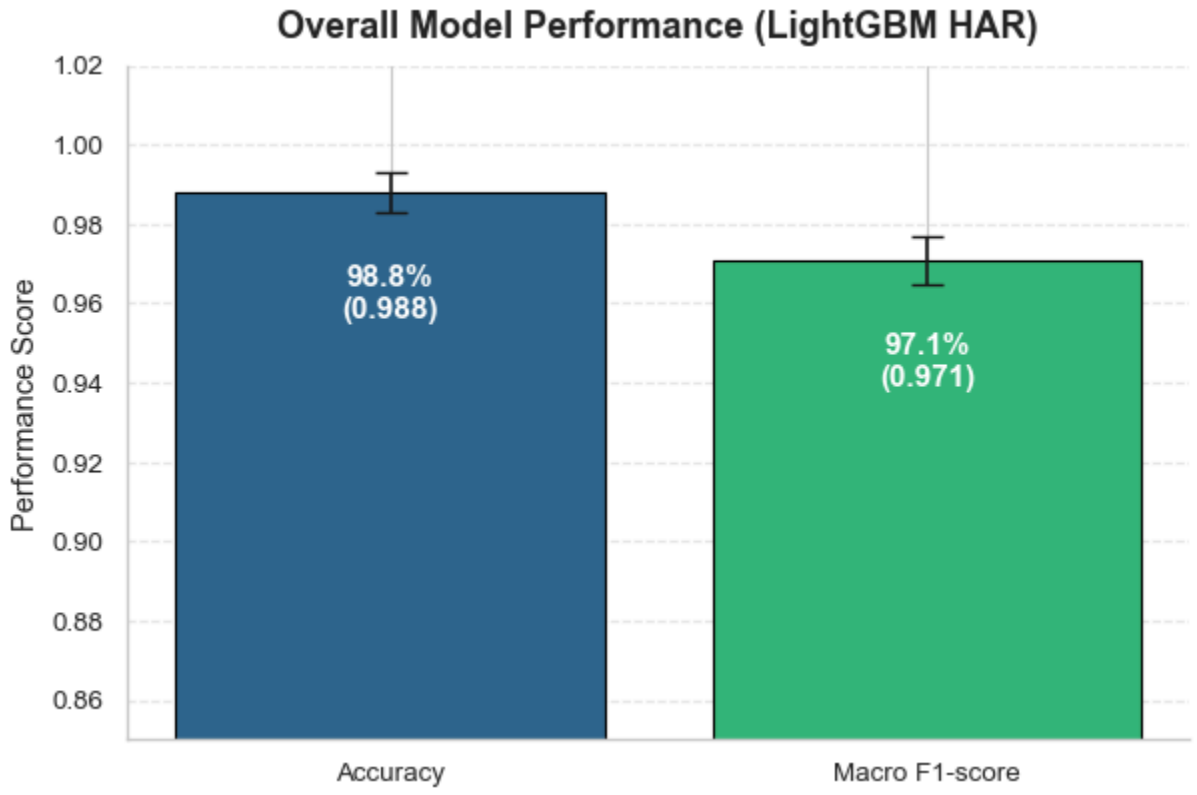

Figure 11: Overall Model Performance

Figure 11 presents the overall performance of the baseline LightGBM model evaluated on the test set using two complementary metrics: accuracy and macro F1-score.

Overall, this result demonstrates that although the feature-based LightGBM model provides a reasonable baseline, it struggles to generalize across all activity classes in a subject-independent setting. This observation motivates the exploration of more expressive models, such as deep learning architectures, and the use of class-aware evaluation metrics when assessing HAR systems.

6.2 Confusion Matrix

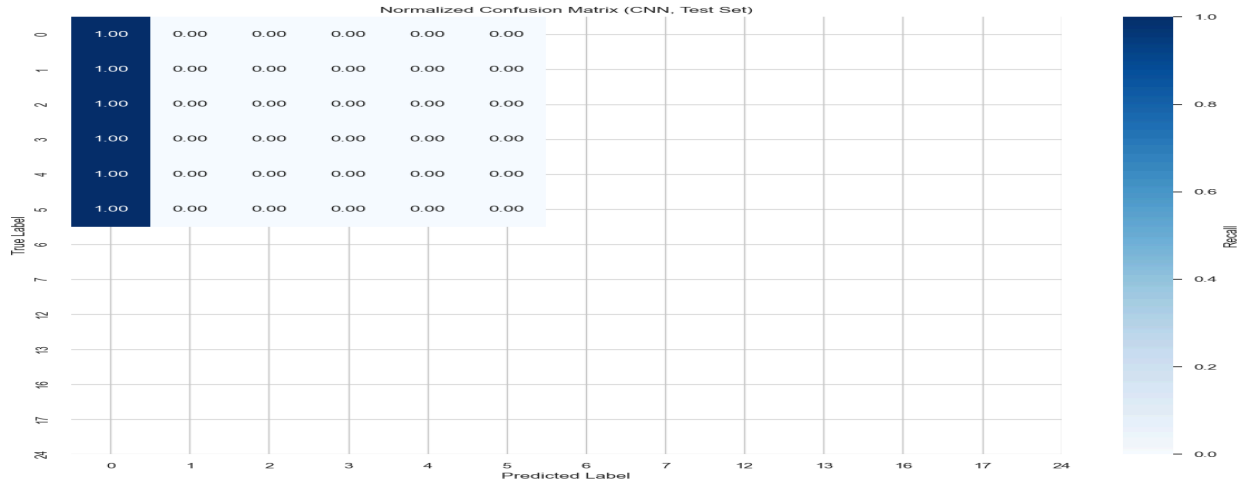

Figure 12: Confusion Matrix

Figure 12 presents the row-normalized confusion matrix of the CNN model on the test set, along with per-class recall scores. Strong diagonal dominance is observed for frequent activities, while reduced recall is evident for minority and transitional classes. This imbalance-driven behavior explains the observed gap between Weighted-F1 and Macro-F1 scores, highlighting the importance of macro-averaged metrics for fair evaluation.

6.3 Per-class breakdown

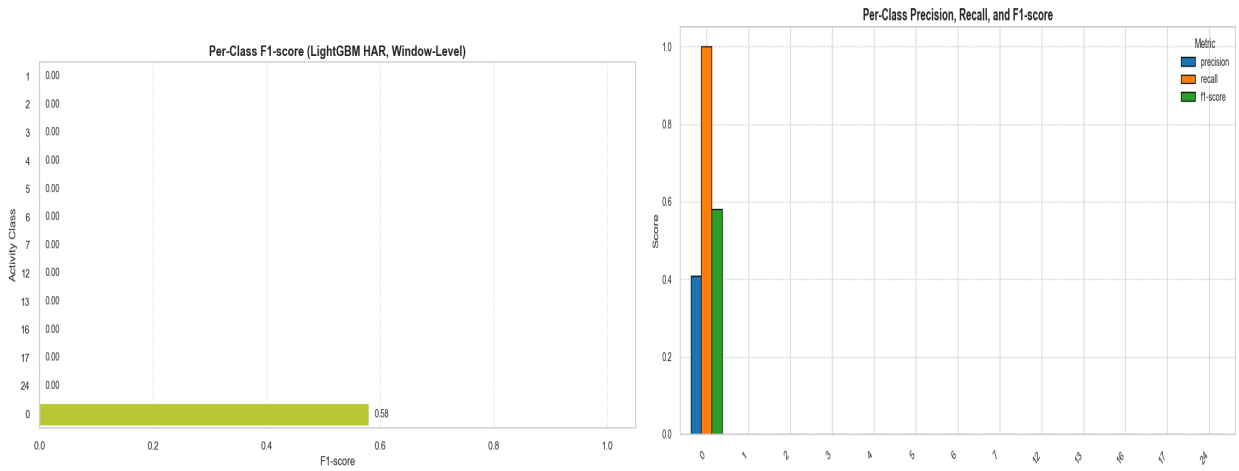

Figure 13: Per-class Breakdown

7. Conclusion

In this project, a subject-independent human activity recognition system was developed using physiological and motion sensor time-series data. To ensure reliable evaluation, a rigorous preprocessing pipeline was employed, including removal of transient activities, sliding-window segmentation, feature extraction, normalization, and subject-wise train, validation, and test splits to prevent data leakage. This setup reflects realistic deployment scenarios in which activity recognition models must generalize to unseen individuals.

A classical feature-based baseline was first established using statistical descriptors extracted from sensor windows and an ensemble classifier. This approach provided a strong and interpretable reference, demonstrating that handcrafted features capture meaningful motion characteristics for activity recognition. Building upon this baseline, a convolutional neural network was explored to directly model temporal dependencies in raw sensor signals. The CNN demonstrated improved performance, particularly in terms of macro-averaged F1-score, indicating better recognition of less frequent and more complex activities.

Comprehensive evaluation using accuracy, macro F1-score, and confusion matrix analysis revealed the impact of class imbalance and highlighted common sources of misclassification among temporally similar activities. The comparison between baseline and deep learning models illustrates the trade-off between interpretability and representational power, and confirms the importance of temporal modeling for time-series HAR tasks.

Overall, the results show that combining careful data preparation with both feature-based and deep learning approaches leads to robust performance in subject-independent human activity recognition. Future work could explore more advanced temporal models such as recurrent or attention-based architectures, sensor fusion strategies, and the incorporation of subject demographic information to further improve generalization and robustness.