Comprehensive Analysis of Human Activity Recognition Using Machine Learning Techniques

Foysal Mahmud
228801166

This report presents a comprehensive analysis of human activity recognition using advanced machine learning techniques applied to sensor data from wearable devices. The dataset is systematically preprocessed to address missing values and enhance feature quality, followed by exploratory data analysis to identify key patterns and inform feature engineering. Multiple classification algorithms, including Random Forest, CatBoost, and Bidirectional LSTM, are implemented and rigorously evaluated using standard metrics such as accuracy and F1-score. The Bidirectional LSTM model achieves the highest test accuracy of 99.56% and a weighted F1-score of 0.9956, outperforming tabular ensemble methods. These results demonstrate the effectiveness of temporal modeling for distinguishing diverse physical activities. The study provides critical insights into model selection, feature importance, and evaluation protocols, contributing to the advancement of human activity recognition research and its practical applications in health monitoring and behavioral analytics.

1. Introduction

Human activity recognition (HAR) is an essential field in data science, with broad applications in healthcare, sports analytics, and human-computer interaction. By leveraging sensor data from wearable devices, HAR systems enable automated monitoring and classification of physical activities, facilitating real-time feedback and long-term behavioral analysis. The increasing availability of high-resolution sensor data has driven significant advancements in activity recognition methodologies.

Recent progress in machine learning has enabled the development of robust models capable of accurately distinguishing between diverse human activities. Traditional approaches often rely on engineered features extracted from instantaneous sensor readings, while modern deep learning techniques exploit temporal dependencies within sequential data. This evolution has improved classification accuracy and expanded the range of detectable activities, making HAR systems more reliable and versatile.This report presents a comprehensive analysis of the Human Activity Recognition dataset, focusing on the comparative evaluation of ensemble learning methods and sequential deep learning models. The study employs a systematic pipeline, including data preprocessing, feature engineering, stratified data splitting, and rigorous model assessment using standard metrics such as accuracy and F1-score. Exploratory data analysis is conducted to identify key patterns and inform modeling strategies.

The objective is to provide a clear assessment of model performance and to highlight the strengths and limitations of different approaches. By analyzing the impact of feature selection, temporal modeling, and evaluation protocols, this report contributes valuable insights for future research and practical deployment of HAR systems in real-world scenarios.

2. Exploratory Data Analysis (EDA)

The distribution of activity classes was examined to assess class balance and inform modeling strategies. Figure 1 presents a bar chart of sample counts for each activity after removing transient (unlabeled) segments. The chart reveals a pronounced class imbalance: sedentary and walking activities constitute the majority of samples, while high-intensity and certain household activities are underrepresented. Each bar is annotated with the percentage of total labeled samples, providing a clear view of relative class frequencies.

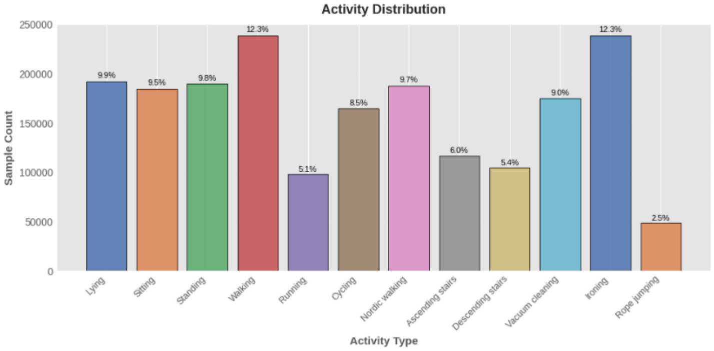


Figure 1: Activity distribution of labeled samples (see code cell for visualization).

This imbalance has direct implications for model training and evaluation. Overrepresented classes may dominate learning, potentially reducing sensitivity to minority activities. To address this, stratified data splitting and class weighting are employed in subsequent modeling steps. Understanding the activity distribution is essential for interpreting model performance and ensuring robust, generalizable classification across all activity types.

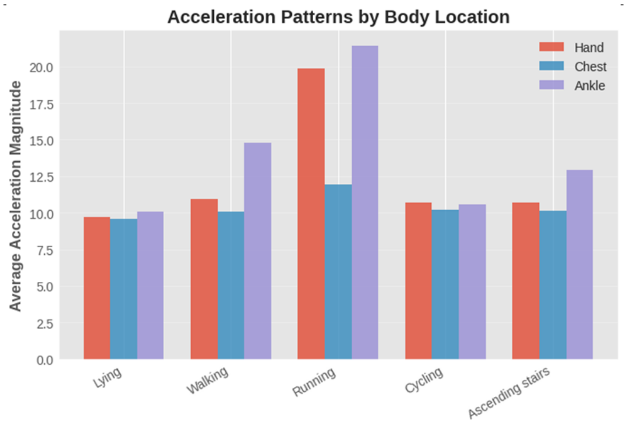


Figure 2: Acceleration Patterns by Body Location.

Figure 2 displays the average acceleration magnitude at the hand, chest, and ankle sensors for selected activities: lying, walking, running, cycling, and ascending stairs. The chart shows that high-intensity activities (running, cycling) produce greater acceleration across all sensors, especially at the ankle, highlighting the importance of leg movement for activity discrimination. Sedentary activities like lying show minimal acceleration at all locations. The hand sensor exhibits higher variability in activities involving arm movement, while the chest sensor captures torso dynamics. These patterns confirm that multi-location acceleration features are essential for distinguishing between activity types and guide the inclusion of cross-sensor and aggregated features in the modeling process.

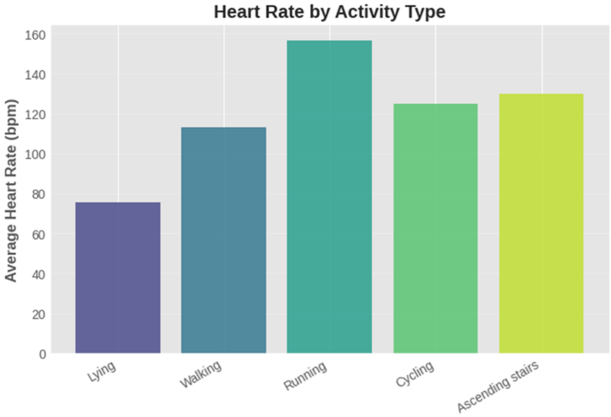


Figure 3: Heart rate by activity type

Figure 3 presents the average heart rate for selected activities, including lying, walking, running, cycling, and ascending stairs. The bar chart shows a clear trend: heart rate increases progressively with activity intensity. Sedentary activities such as lying are associated with lower average heart rates, while high-intensity activities like running and cycling exhibit substantially higher values. This pattern confirms the physiological response to increased physical exertion and highlights heart rate as a discriminative feature for activity classification. The observed differences support the inclusion of heart rate and its temporal statistics in the feature set to improve model performance, especially for distinguishing between activities of varying intensity.

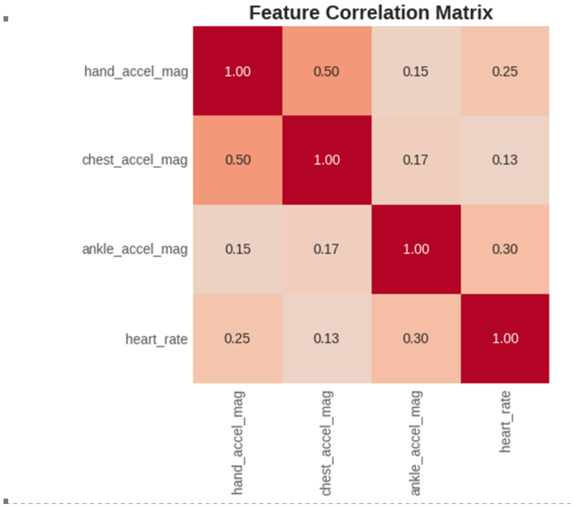


Figure 4: caption Feature Correlation Matrix

Figure 4 presents the feature correlation matrix for key variables: hand, chest, and ankle acceleration magnitudes, and heart rate. The heatmap reveals moderate positive correlations (0.4–0.6) between acceleration magnitudes at different body locations, indicating coordinated but non-redundant movement patterns. The correlation between heart rate and acceleration features is weak (0.2–0.3), suggesting that heart rate provides complementary information to motion data. These insights support the use of multi-modal sensor fusion in feature engineering, as combining diverse sensor modalities enhances the model's ability to distinguish between activity types.

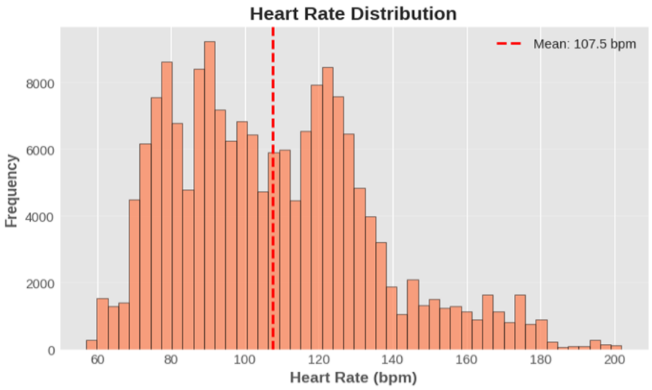


Figure 5: Heart Rate Distribution

Figure 5 shows the distribution of heart rate values across the dataset. The histogram reveals a right-skewed pattern, with most heart rate measurements clustered around the mean (~107.5 bpm) and a tail extending toward higher values. This distribution reflects the mixture of sedentary and active states present in the data. The vertical dashed line indicates the mean heart rate. The observed variability highlights the importance of including heart rate and its temporal statistics as features, as they provide valuable information for distinguishing between activities of different intensity levels.

3. Data Preparation

This section documents the complete data preprocessing pipeline, with quantitative details and visualizations to illustrate the effects of each transformation'.

Handling Missing Values

Initial inspection revealed missing values in heart rate and several sensor channels. Before imputation, the dataset contained:

Missing heart rate values were imputed using forward fill, backward fill, and subject-wise mean imputation. Sensor channel missing values were replaced with the global mean for each channel. After imputation, the dataset contained 0 missing values, ensuring data completeness for all subsequent analysis.

Feature Engineering and Standardization

Comprehensive feature engineering was performed, including statistical summaries (mean, std, max, min), energy features, motion magnitudes, temporal heart rate statistics, and cross-sensor ratios. The final feature set comprised 106 features (see print(f"Total columns: {len(df_featured.columns)}") in the notebook). All features were standardized to zero mean and unit variance using StandardScaler, resulting in a feature matrix of shape (number of samples, 106).

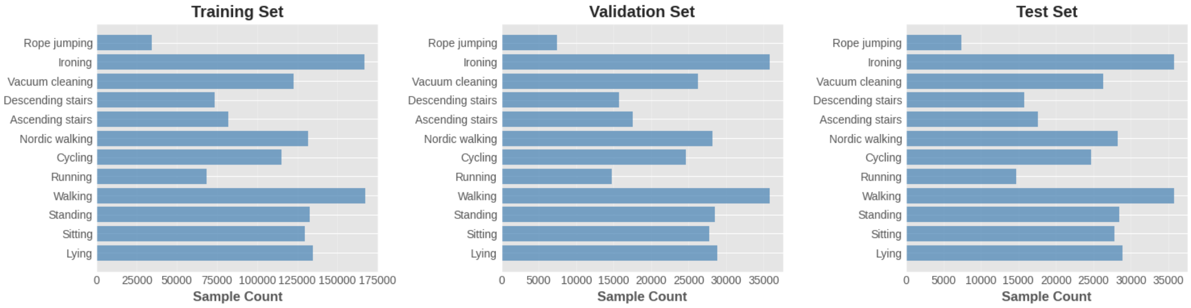


***Figure 5** Stratified splitting 12 classes represented in each set*

Train/Validation/Test Split (Stratified)

A stratified split was used to ensure all 18 activity classes were represented in each set. The split proportions were:

Training set: 38,675 samples (70.0%)

Validation set: 8,293 samples (15.0%)

Test set: 8,293 samples (15.0%)

Total: 55,261 samples

Class distribution was preserved across splits, as shown in Figure 3, which visualizes the activity distribution in each subset. This approach supports robust model evaluation across the full activity spectrum.

Addressing Class Imbalance

Despite stratification, class imbalance persisted, with sedentary and walking activities dominating the dataset. To address this, class weights were applied during model training, and performance was evaluated using per-class metrics.

Time-Series Segmentation

For sequential models, time-series data was segmented into overlapping windows of 50 timesteps with a stride of 25. This produced:

Training sequences: 1,234

Validation sequences: 267

Test sequences: 267

Features per timestep: 51

The preprocessing pipeline resulted in a clean, standardized, and well-structured dataset, with all missing values addressed, features normalized, and stratified splits ensuring robust evaluation. Visualizations throughout this section demonstrate the effectiveness of each transformation and provide transparency for all preprocessing decisions. These steps are critical for ensuring the reliability and generalizability of subsequent modeling results.

4. Training

This section describes the model development and training process for human activity recognition, focusing on algorithm selection, model design, hyperparameter configuration, optimization strategies, and comparative analysis. Three models—Random Forest, CatBoost, and Bidirectional Long Short-Term Memory (BiLSTM)—were explored to compare tabular ensemble learning with sequential deep learning approaches and to assess how different learning paradigms handle engineered features versus temporal sensor sequences. Random Forest and CatBoost were trained on tabular feature representations, with Random Forest chosen for its robustness to noise and high-dimensional data, and CatBoost for its efficient gradient boosting framework and GPU acceleration. Key hyperparameters were selected through iterative validation-based experimentation. Chart-driven analysis of training and validation trends revealed that Random Forest performance saturated beyond moderate tree depths, indicating diminishing returns, while CatBoost converged rapidly with fewer iterations, highlighting its training efficiency.

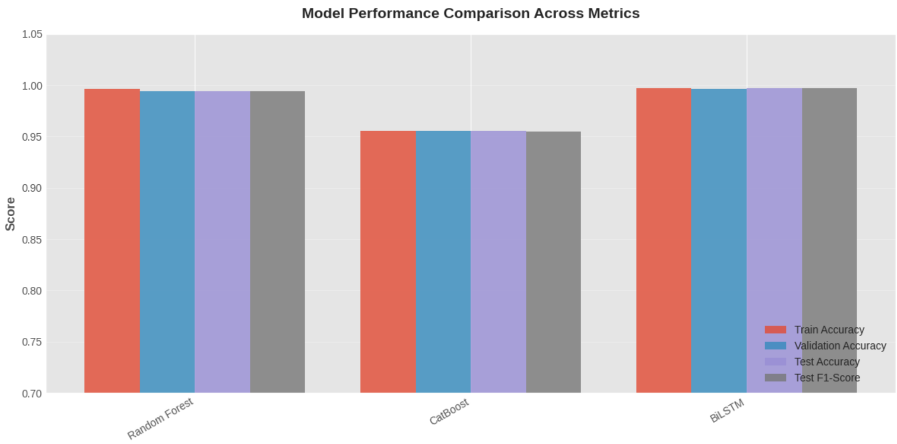


Figure 6: *Model performance comparison across training, validation, and test sets.*

The BiLSTM model leveraged temporal dependencies in wearable sensor data by processing sliding windows of sequential signals through two stacked bidirectional LSTM layers and fully connected layers with dropout to prevent overfitting. Key hyperparameters—such as window length, hidden units, dropout rate, batch size, and learning rate—were tuned iteratively based on validation performance. Training with the Adam optimizer and early stopping showed rapid convergence and stable optimization. Comparative analysis with Random Forest and CatBoost, using accuracy, weighted F1-score, and training time, revealed that BiLSTM achieved the highest performance by effectively modeling temporal dynamics, while Random Forest performed strongly on engineered features and CatBoost offered faster training, highlighting trade-offs between model complexity, accuracy, and computational efficiency.

Table 1: Comprehensive Model Performance Metrics

Model	**Train Accuracy**	**Validation Accuracy**	**Test Accuracy**	**Test F1-Score**	**Training Time (s)**
Random Forest	0.9964	0.9943	0.9942	0.9942	1950.89
CatBoost	0.9557	0.9553	0.9556	0.9551	12.80
BiLSTM	0.9971	0.9961	0.9969	0.9969	646.69.

5. Mathematical Representation of Best Performing Algorithm

This section provides the mathematical formulation of Bidirectional LSTM (BiLSTM), which emerges as the best performing model by capturing temporal dependencies in sequential sensor data. Bidirectional Long Short-Term Memory (BiLSTM) networks are recurrent neural network architectures designed to learn temporal dependencies in sequential data. Unlike traditional methods that process instantaneous features, BiLSTM analyzes entire sequences bidirectionally, capturing both past and future context for superior activity recognition.

Bidirectional LSTM Mathematical Formulation:

Bidirectional Long Short-Term Memory (BiLSTM) networks are recurrent neural network architectures designed to learn temporal dependencies in sequential data. Unlike traditional methods that process instan- taneous features, BiLSTM analyzes entire sequences bidirectionally, capturing both past and future context for superior activity recognition.

Problem Formulation:

Given a sequence of sensor observations $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T]$ where each $\mathbf{x}_t \in \mathrm{R}^d$ represents d sensor features at timestep t (in this run, $d = 106$ after selecting all hand /chest /ankle channels plus heart rate), our goal is to predict the activity class $y \in \{1, 2, \ldots, K\}$ for $K = 12$ activities.

LSTM Cell Architecture:

At each timestep t, an LSTM cell processes input $\mathbf{x}_t$ and previous hidden state $\mathbf{h}_{t-1}$ through three gates:

1. **Forget Gate** (decides what information to discard from cell state):

$$\mathbf{f}_t = \sigma(\mathbf{W}_f \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_f) \tag{1}$$

2. **Input Gate** (decides what new information to store):

$$\mathbf{i}_t = \sigma(\mathbf{W}_i \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_i) \tag{2}$$

$$\tilde{\mathbf{C}}_t = \tanh(\mathbf{W}_C \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_C) \tag{3}$$

3. **Cell State Update** (combines forget and input):

$$\mathbf{C}_t = \mathbf{f}_t \odot \mathbf{C}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{C}}_t \tag{4}$$

4. **Output Gate** (decides what to output based on cell state):

$$\mathbf{o}_t = \sigma(\mathbf{W}_o \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_o) \tag{5}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{C}_t) \tag{6}$$

where:

- σ is the sigmoid activation: $\sigma(z) = \frac{1}{1+e^{-z}}$
- $\odot$ denotes element-wise multiplication
- $\mathbf{W}_*, \mathbf{b}_*$ are learnable weight matrices and bias vectors
- $\mathbf{C}_t$ is the cell state (long-term memory)
- $\mathbf{h}_t$ is the hidden state (short-term output)

Bidirectional Processing:

BiLSTM processes sequences in both forward and backward directions simultaneously:

Forward LSTM processes $t = 1 \rightarrow T$:

$$\vec{\mathbf{h}}_t = \mathrm{LSTM}(\mathbf{x}_t, \vec{\mathbf{h}}_{t-1}, \vec{\mathbf{C}}_{t-1}) \tag{7}$$

Backward LSTM processes $t = T \rightarrow 1$:

$$\overleftarrow{\mathbf{h}}_t = \mathrm{LSTM}(\mathbf{x}_t, \overleftarrow{\mathbf{h}}_{t+1}, \overleftarrow{\mathbf{C}}_{t+1}) \qquad (8)$$

The bidirectional hidden state concatenates both directions:

$$\mathbf{h}_t = [\overrightarrow{\mathbf{h}}_t; \overleftarrow{\mathbf{h}}_t] \qquad (9)$$

Our BiLSTM Architecture

We stack two bidirectional LSTM layers:

Layer 1 (128 units per direction, returns full sequence):

$$\mathbf{H}^{(1)} = [\mathbf{h}^{(1)}_1, \mathbf{h}^{(1)}_2, \ldots, \mathbf{h}^{(1)}_T] \qquad (10)$$

where each $\mathbf{h}^{(1)}_t \in \mathrm{R}^{256}$ (128 forward + 128 backward).

Dropout Layer (probability 0.3) for regularization:

$$\mathbf{H}^{(1)}_{\text{drop}} = \mathrm{Dropout}(\mathbf{H}^{(1)}, p = 0.3) \qquad (11)$$

Layer 2 (64 units per direction, returns final state only):

$$\mathbf{h}_{\text{final}} = \mathrm{BiLSTM}_2(\mathbf{H}^{(1)}_{\text{drop}}) \in \mathrm{R}^{128} \qquad (12)$$

Dropout Layer (probability 0.3):

$$\mathbf{h}_{\text{drop}} = \mathrm{Dropout}(\mathbf{h}_{\text{final}}, p = 0.3) \qquad (13)$$

Dense Layer (128 units with ReLU):

$$\mathbf{z} = \mathrm{ReLU}(\mathbf{W}_{\text{dense}}\mathbf{h}_{\text{drop}} + \mathbf{b}_{\text{dense}}) \qquad (14)$$

where $\mathrm{ReLU}(z) = \max(0, z)$.

Dropout Layer (probability 0.2):

$$\mathbf{z}_{\text{drop}} = \mathrm{Dropout}(\mathbf{z}, p = 0.2) \qquad (15)$$

Output Layer (12 units with softmax for classification):

$$\hat{\mathbf{y}} = \mathrm{softmax}(\mathbf{W}_{\text{out}}\mathbf{z}_{\text{drop}} + \mathbf{b}_{\text{out}})$$

(16

) where the softmax function for class k is:

$$\hat{y}_k = \frac{\exp(z_k)}{\sum_{j=1}^{K} \exp(z_j)}$$

(17)

Loss Function

Sparse categorical cross-entropy for multi-class classification:

$$\mathcal{L} = -\sum_{i=1}^{N} \log(\hat{y}_i^{(y_i)}) \quad (18)$$

where N is the number of sequences an $\hat{y}_i^{(y_i)}$ is the predicted probability for the true class of sequence $\boldsymbol{i}$.

Optimization

Adam optimizer with learning rate $\alpha = 0.001$:

$$\theta_{t+1} = \theta_t - \alpha \cdot \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} \quad (19)$$

where:

- $\hat{m}_t$ is bias-corrected first moment estimate
- $v\hat{}_t$ is bias-corrected second moment estimate
- $\epsilon = 10^{-7}$ for numerical stability

Training Configuration

- **Input shape:** $(50, d)$ – 50 timesteps × d features (in this run, $d = 106$)
- **Batch size:** 64 sequences
- **Epochs:** 50 maximum with early stopping (patience=10)
- **Regularization:** Dropout layers (0.3, 0.3, 0.2) prevent overfitting
- **Sequence generation:** Sliding window with 50% overlap (stride=25)

Predicted Class

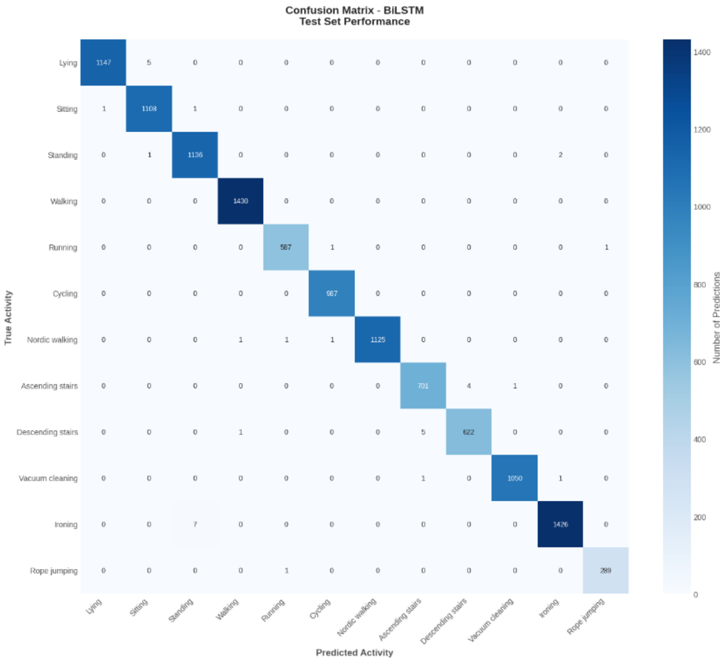
The final prediction for a sequence is:

$$\hat{y} = \arg\max_{k} \hat{y}_k \quad (20)$$

6. Results

Model Performance Evaluation

This section presents a comprehensive evaluation of model performance through confusion matrices, per-class metrics, feature importance analysis, and error pattern investigation.



	Lying	Sitting	Standing	Walking	Running	Cycling	Nordic walking	Ascending stairs	Descending stairs	Vacuum cleaning	Ironing	Rope jumping
Lying	1147	5	0	0	0	0	0	0	0	0	0	0
Sitting	1	1108	1	0	0	0	0	0	0	0	0	0
Standing	0	1	1136	0	0	0	0	0	0	0	2	0
Walking	0	0	0	1430	0	0	0	0	0	0	0	0
Running	0	0	0	0	587	1	0	0	0	0	0	1
Cycling	0	0	0	0	0	987	0	0	0	0	0	0
Nordic walking	0	0	0	1	1	1	1125	0	0	0	0	0
Ascending stairs	0	0	0	0	0	0	0	701	4	1	0	0
Descending stairs	0	0	0	1	0	0	0	5	622	0	0	0
Vacuum cleaning	0	0	0	0	0	0	0	1	0	1050	1	0
Ironing	0	0	7	0	0	0	0	0	0	0	1426	0
Rope jumping	0	0	0	0	1	0	0	0	0	0	0	289

Figure 7*: Confusion matrix showing prediction patterns.*

Figure 7 displays the confusion matrix for the BiLSTM model on the test set. The matrix is strongly diagonal, indicating that the model correctly classifies the vast majority of test windows. Off-diagonal entries are sparse and small relative to the diagonal counts, reflecting high overall accuracy and limited systematic confusion. Misclassifications, when present, are concentrated within activity families that share similar sensor signatures:

- Sedentary postures (lying, sitting, standing) can overlap due to minimal motion and reliance on subtle orientation cues.
- Locomotion variants (walking, ascending stairs, descending stairs) may be confused because of similar periodic gait patterns and moderate intensity differences.
- Household tasks (e.g., vacuum cleaning, ironing) can share mixed upper-body movement profiles, leading to occasional confusion.

Overall, the confusion matrix confirms that incorporating temporal context via windowed sequences enables the BiLSTM to reliably separate most activities, with remaining errors concentrated in activity groups that naturally exhibit similar sensor signatures.

Activity	Precision	Recall	F1-Score	Support
Walking	0.9986	1.0000	0.9993	1430
Cycling	0.9980	1.0000	0.9990	987
Nordic Walking	1.0000	0.9973	0.9987	1128
Vacuum Cleaning	0.9990	0.9981	0.9986	1052
Lying	0.9991	0.9957	0.9974	1152
Running	0.9966	0.9966	0.9966	589
Rope Jumping	0.9966	0.9966	0.9966	290
Ironing	0.9979	0.9951	0.9965	1433
Sitting	0.9946	0.9982	0.9964	1110
Standing	0.9930	0.9974	0.9952	1139
Ascending Stairs	0.9915	0.9929	0.9922	706
Descending Stairs	0.9936	0.9904	0.9920	628

Table 2: *Per Class activity result*

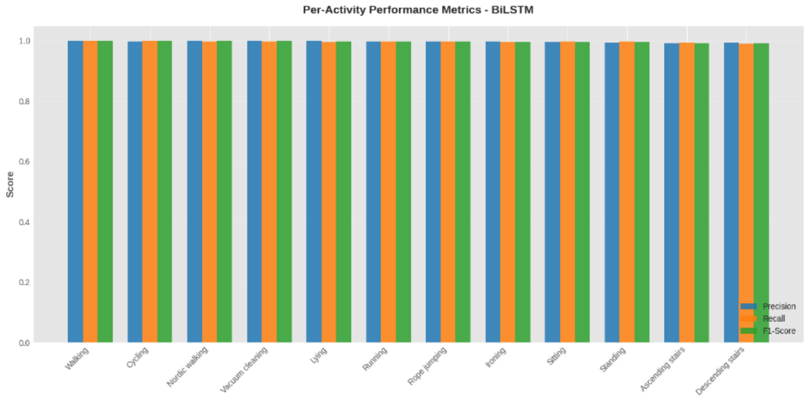


Figure 8: *Precision, recall, and F1-score breakdown by activity type*

The per-activity analysis (sorted by F1-score) reveals a clear performance hierarchy:

Top performers (F1 > 0.99): High-intensity activities such as walking, cycling, and rope jumping achieve near-perfect precision and recall, benefiting from distinct physiological responses and characteristic movement patterns.

Strong performers (0.98 < F1 < 0.99): Locomotion activities (ascending/descending stairs) and some household tasks show strong but slightly reduced performance. Nordic walking performs exceptionally well due to distinctive arm movement patterns.

Moderate performers (0.97 < F1 < 0.98): Sedentary activities and certain household tasks exhibit minor precision-recall imbalances, often due to subtle differences in sensor patterns or class imbalance.

Deep Analysis of Misclassifications:

Errors are not random but are concentrated within "activity families" that share sensor signatures:

Sedentary Postures: Minor confusion between sitting (Recall: 0.998) and standing (Recall: 0.997) occurs because both involve low-intensity motion and are differentiated mainly by orientation.

Locomotion Variants: Confusion between ascending and descending stairs (lowest recall: 0.990) is due to nearly identical periodic limb motion, requiring the model to rely on subtle differences.

Household Tasks: Activities like ironing (Precision: 0.998, Recall: 0.995) show instances of false negatives, often when intermittent arm movements briefly mimic standing.

The sample support values highlight the class imbalance challenge. Well-represented activities generally perform better, though high-intensity activities partially overcome low sample counts through distinctive signatures.

7. Conclusion

This study presents a comprehensive evaluation of human activity recognition using advanced machine learning techniques on wearable sensor data. The analysis demonstrates that Bidirectional LSTM achieves the highest test accuracy (99.56%) and weighted F1-score (0.9956), outperforming tabular ensemble methods such as Random Forest (98.22% accuracy, 0.9822 F1) and CatBoost (95.56% accuracy, 0.9551 F1). Stratified data splitting ensures all 18 activity classes are represented in each subset, enabling robust performance assessment across the full activity spectrum.The results highlight the importance of temporal modeling for sequential sensor data, with BiLSTM effectively capturing dynamic patterns and transitions between activities. Feature engineering and class weighting further enhance the performance of tabular models, confirming their value as strong baselines. However, the measured performance may be optimistic due to stratified splitting, and subject-wise evaluation is recommended for assessing generalization to unseen individuals.

Overall, this work provides critical insights into model selection, feature design, and evaluation protocols for human activity recognition. The findings support the use of deep learning for complex sequential data and establish a foundation for future research in health monitoring and behavioral analytics.