Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2026.074057

ARTICLE

A Hybrid Deep Learning Approach for IoT-Enabled Human Activity Recognition and Advanced Analytics

Shtwai Alsubai[1], Abdullah Al Hejaili[2], Najib Ben Aoun[3,4,*], Amina Salhi[5] and Vincent Karovič[6,*]

[1]College of Computer Engineering and Sciences, Prince Sattam bin Abdulaziz University, AlKharj, Saudi Arabia
[2]Faculty of Computers & Information Technology, Computer Science Department, University of Tabuk, Tabuk, Saudi Arabia
[3]Faculty of Computing and Information, Al-Baha University, Alaqiq, Saudi Arabia
[4]REGIM-Lab: Research Groups in Intelligent Machines, National School of Engineers of Sfax (ENIS), University of Sfax, Sfax, Tunisia
[5]Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh, Saudi Arabia
[6]Department of Information Management and Business Systems, Faculty of Management, Comenius University Bratislava, Odbojárov 10, Bratislava, Slovakia

*Corresponding Authors: Najib Ben Aoun. Email: najib.benaoun@ieee.org; Vincent Karovič. Email: vincent.karovic6@fm.uniba.sk

Received: 30 September 2025; Accepted: 16 December 2025; Published: 12 March 2026

ABSTRACT: The concept of Human Activity Recognition (HAR) is integral to applications based on Internet of Things (IoT)-enabled devices, particularly in healthcare, fitness tracking, and smart environments. The streams of data from wearable sensors are rich in information, yet their high dimensionality and variability pose a significant challenge to proper classification. To address this problem, this paper proposes hybrid architectures that integrate traditional machine learning models with a deep neural network (DNN) to deliver improved performance and enhanced capabilities for HAR tasks. Multi-sensor HAR data were used to systematically test several hybrid models, including: RF + DNN (Random Forest + Deep Neural Network), XGB + DNN (XGBoost + DNN), GB + DNN (Gradient Boosting + DNN), KNN + DNN (K-Nearest Neighbors + DNN), and DT + DNN (Decision Tree + DNN). The RF + DNN model was the most accurate, achieving a 97.03% score with excellent precision, recall, and F1-score. These findings demonstrate that hybrid machine learning and deep learning systems have a promising future in IoT-based HAR applications. The model provides a novel solution for developing smart and trustworthy monitoring systems that support real-time analytics, patient surveillance, and other IoT applications.

KEYWORDS: Human activity recognition (HAR); Internet of Things (IoT); wearable sensors; hybrid models; deep neural networks (DNN)

1 Introduction

The Internet of Things (IoT) has transformed how contemporary societies engage with technology by establishing a ubiquitous system of interconnected sensors capable of processing, transmitting, and sensing data in real time [1]. The IoT is transforming everyday life, with applications in healthcare, transportation, smart cities, and industrial automation that enable autonomous decisions, optimize resources, and facilitate predictive analytics [2]. One key feature of IoT ecosystems is that they generate continuous, heterogeneous, and high-dimensional data streams from various sensors in smartphones, wearables, and other bright objects [3]. The critical aspect is processing the immense volume of sensor data to unlock the full potential of IoT and deliver intelligent, adaptive services. HAR can be considered one of the most powerful IoT-based

applications, with applications in personalized healthcare, elderly care, physical rehabilitation, fitness tracking, workplace safety, and home automation [4]. Motion sensors supported by HAR (e.g., accelerometers, gyroscopes, and magnetometers) are used to understand not only simple behaviors (e.g., walking, running, sitting) but also more complex ones (e.g., exercising, cooking, driving) [5].

Over the past several years, deep learning has demonstrated remarkable potential in HAR through providing automatic feature extraction and hierarchical representation learning [6–8]. Convolutional Neural Networks (CNNs) can be used to model spatial patterns. In contrast, Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) and Attention-based networks are more effective at modeling temporal dynamics. Although these single deep learning models have their benefits, they can be limited by their generalization capabilities and their ability to harness only spatial and temporal correlations. These shortcomings highlight the importance of hybrid deep learning models that combine complementary neural networks to achieve greater recognition accuracy and robustness. In practice, convolutional layers can effectively extract local spatial features, while LSTM or attention mechanisms can model long-term temporal dependencies. Integrating these elements into a single framework enables more detailed representations of activities and enhances flexibility and scalability for real-world IoT contexts.

- The paper proposes a hybrid architecture that combines strategically selected classical ensemble learning algorithms (RF, XGB, KNN, GB, DT) with a Deep Neural Network (DNN) to provide the interpretability and decision limits of traditional ML models with the hierarchical feature learning of deep networks. This combination improves accuracy, robustness, and generalization, especially for complex, high-dimensional sensor data from IoT, something that is not always considered in traditional HAR models.
- The proposed method contrasts with previous methods, which merely stack ML and DL implementations; instead, it highlights an IoT-friendly analytical pipeline that emphasizes computational flexibility and scalability across disparate sensor modalities. This emphasis aligns with the practical needs of real-world IoT-based healthcare and intelligent systems for monitoring, where inference efficiency is essential but accuracy is paramount.
- Moreover, we compare various hybrid types (RF + DNN, XGB + DNN, GB + DNN, KNN + DNN, and DT + DNN) and present empirical evidence of complementarity between the models. The RF + DNN outperformed the other ones by achieving the best overall performance (accuracy exceeding 97%), establishing a robust standard for HAR applications based on IoT.

2 Related Work

In [9], a dual-purpose IoT architecture was developed that simultaneously handled localization and HAR in noisy, sensor-rich scenarios. Denoising was performed using a Chebyshev Type-I filter, signal windowing, parallel feature extraction, Boruta feature selection, and PSO optimization, followed by training with two RNNs specifically designed for HAR and localization. Testing on the Extrasensory and SHL data sets demonstrated that the system achieved higher accuracies than state-of-the-art, with HAR accuracies of 89.25% and 95.75%, and localization accuracies of 90.50% and 91.50%. Authors in [10] explored methods in the HAR field, filling the gap between classical algorithms (Fourier Transform, Wavelet Transform, PCA) and deep learning methods (CNNs, RNNs, Transformers). It focused on multimodal sensing (accelerometers, gyroscopes, EMG, EEG, thermal, and infrared) and reviewed 217 papers to identify strengths and limitations, including noise resistance, real-time processing, and scalability. The researchers have developed a roadmap that supports multimodal data fusion and lightweight architectures of next-generation HAR systems.

Authors in [11] present a tag-free inside fall detector and utilize a transformer network encoder with data fusion methods. The purpose of the study is to collect received signal strength indicator (RSSI) and phase data to monitor elderly people contactlessly using passive ultra-high-frequency (UHF) RFID tags.

The superior transformer model's ability to balance modelling of long-range dependencies with minimal preprocessing enables the proposed framework to improve the accuracy of activity recognition and fall detection. The technique has good performance relative to the traditional deep learning methodologies like CNN, RNN and LSTM and is also reliable beyond a 3-m distance. This is indicative of the approach's potential for practical, low-cost, and non-invasive implementation in real-life settings of elderly care. Authors in [12] proposed a hybrid transformer framework for accurately identifying human activity using consumer electronic devices. To overcome the computational bottlenecks of these devices, the proposed model uses a low-weight MobileNetV3 to learn salient spatiotemporal features, and a residual-based Transformer Network (SRTN) to learn long-range temporal dependencies. The information refinement and reduction of irrelevant information, caused by the SRTN residual connections and the multi-head self-attention mechanism, lead to a more efficient representation of video sequences. They conducted experiments on three standard HAR datasets showing that the proposed framework is stronger and more efficient, achieving 76.14%, 96.63%, and 97.31%, respectively, thereby demonstrating its effectiveness.

To address the limited real-world fall data, the Authors in [13] proposed a GAN-enhanced IoT system for fall detection and HAR. GAN-generated real fall occurrences were added to the training data, and a 1D CNN was used to extract features from accelerometer and gyroscope data. The system achieved significant fall detection accuracy, with a low number of false positives, and was able to classify 15 types of falls and 19 types of daily activities. In the context of elder care, the Authors in [14] present an IoT-based HAR system that leverages preprocessing, the GRU model, and federated distillation for privacy-preserving monitoring. GRU networks were used to process filtered, transformed raw sensor data, and knowledge sharing was decentralized via federated learning. The system was found to be 95% accurate, with an F1-score of 0.94, demonstrating its usefulness for scalable, privacy-sensitive HAR implementations. In [15], Chatty Factories were proposed, in which products with IoT connectivity actively report usage to designers and manufacturers. A prototype was used to gather sensor data on six activities each day, and after preprocessing, labeling, and clustering using four unsupervised algorithms. Fuzzy C-Means yielded the best result, with an F-measure of 0.87 and an MCC of 0.84, indicating that unsupervised HAR can be utilized to achieve Industry 4.0 product-use analytics. Authors in [16] introduced an ensemble learning model for HAR by combining multiple classifiers with HAR sensor data. Following the preprocessing step, base classifiers were trained and optimized KNN, Decision trees and Random Forests before majority voting was used to combine predictions. The four HAR datasets studied (WISDM, HAPT, HAR, and KU-HAR) have demonstrated better performance in terms of accuracy, precision, recall, and F1-score, highlighting the potential of ensemble methods. Table 1 shows the summary of the discussed studies:

Table 1: Summary of recent studies related to IoT-based human activity recognition (HAR)

Ref.	Objective/Focus	Techniques/Methods used	Datasets	Performance/Key findings
[9]	Dual-purpose IoT architecture for localization and HAR in noisy environments	Chebyshev Type-I filter, signal windowing, parallel feature extraction, Boruta feature selection, PSO optimization, and RNN-based HAR/localization	Extrasensory, SHL	HAR: 89.25%, 95.75%; Localization: 90.50%, 91.50%.

(Continued)

Table 1 (continued)

Ref.	Objective/Focus	Techniques/Methods used	Datasets	Performance/Key findings
[13]	GAN-enhanced IoT-based fall detection and HAR system with limited real data	GAN-generated fall data, 1D CNN using accelerometer and gyroscope signals	Custom fall detection dataset	Detected 15 fall types and 19 activities with high accuracy, low false positives.
[14]	Privacy-preserving IoT-based HAR system for elder care using federated learning	Preprocessing, GRU model, federated distillation for decentralized privacy-aware training	Elder care IoT dataset	Accuracy: 95%, F1-score: 0.94.
[16]	Ensemble learning model for HAR classification	Preprocessing, base classifiers (KNN, DT, RF), majority voting	WISDM, HAPT, HAR, KU-HAR	Improved performance demonstrates the effectiveness of ensemble methods.

3 Dataset and System Design

The Mobile Health (MHEALTH) dataset [https://archive.ics.uci.edu/dataset/319/mhealth+dataset] contains detailed measurements of body movement and vital signs for ten volunteers with diverse profiles, collected during 12 physical activities performed at varying times and durations. They are not only static exercises, such as standing, sitting, and lying down, but also dynamic exercises, including walking, climbing stairs, cycling, jogging, running, and jump-based exercises, as presented in Table 2. The MHEALTH dataset was selected because it offers a variety of multimodal physiological and motion cues from wearable sensors, enabling robust assessment of hybrid deep learning models for human activity identification. The comparability and reproducibility of the results can be ensured by its standardized collection process and its widespread use in prior HAR research. To collect data, three Shimmer2 wearable sensors were placed on the chest, right wrist, and left ankle, enabling the measurement of body dynamics at multiple sites. Both sensors captured triaxial acceleration, gyroscope angular velocity, and magnetic field orientation at 50 Hz. The chest-mounted sensor also provided two-lead ECG signals (which were not utilized in this research) that could be used in future healthcare-related applications. The data set was gathered in real-world settings outside the lab, under natural performance conditions, where the intensity, style, and pace of activities varied. The design enables the dataset to be generalizable to real-world daily living conditions, serving as a strong reference point in human activity recognition studies within an IoT-based environment.

3.1 Data Preprocessing

Raw MHEALTH data were first analyzed to assess data quality and structure. Next, a statistical summary of each numerical attribute was generated to examine its range and distribution. We performed data quality checks by identifying and removing null values and duplicates. A sampling strategy was used to address the class imbalance problem, mainly due to the disproportionately high rate of a specific activity (especially the one labeled 0). In particular, 40,000 samples from the dominant activity category were randomly selected and

combined with the remaining activity information, resulting in a more balanced class distribution. Formally, assume that the dataset is given by Eq. (1).

$$D = \{(x_i, y_i) \mid i = 1, 2, \ldots, N\} \tag{1}$$

where $y_i \in \{1, 2, \ldots, K\}$ is the activity label, $x_i \in \mathbb{R}^d$ is the feature vector, and the total number of samples is given by N. The next step was to remove outliers with the help of the quantile range algorithm, by which the data that were not within the range of the lower ($Q_{0.01}$) and upper ($Q_{0.99}$) quantiles were discarded as in Eq. (2).

$$Q_{0.01}(x_j) \leq x_{ij} \leq Q_{0.99}(x_j), \quad \forall j \in \{1, \ldots, d\} \tag{2}$$

Table 2: Description of activities in the MHEALTH dataset

Activity ID	Activity description	Duration/Repetitions
L1	Standing still	1 min
L2	Sitting and relaxing	1 min
L3	Lying down	1 min
L4	Walking	1 min
L5	Climbing stairs	1 min
L6	Waist bends forward	20 repetitions
L7	Frontal elevation of arms	20 repetitions
L8	Knees bending (crouching)	20 repetitions
L9	Cycling	1 min
L10	Jogging	1 min
L11	Running	1 min
L12	Jump front & back	20 repetitions

x_{ij} the value of feature j in instance i. Categorical variables like subject numbers and activity types were coded into numeric values using the Label Encoding function, which is defined in Eq. (3).

$$y'_i = \mathrm{LE}(y_i), \quad y'_i \in \{0, 1, \ldots, K-1\} \tag{3}$$

This change was compatible with downstream machine learning algorithms. Lastly, the Robust Scaler was used to scale the features by calculating the difference between the median and dividing it by the interquartile range (IQR), as in Eqs. (4) and (5).

$$x_j^{\text{scaled}} = \frac{x_j - \text{Median}(x_j)}{IQR(x_j)} \tag{4}$$

$$IQR(x_j) = Q_{0.75}(x_j) - Q_{0.25}(x_j) \tag{5}$$

Here, x_j denotes the feature value, $\text{Median}(x_j)$ is the median of the feature, and $IQR(x_j)$ represents its interquartile range computed as the difference between the 75th and 25th percentiles. This scaling method is beneficial when sensor data are involved, as it minimizes the impact of outliers compared to conventional normalization methods. Based on a 75:25 split, the dataset was divided into training and test sets, ensuring that the models were trained on a sufficiently large subset of data without compromising a

separate subgroup for model evaluation. The processed data were organized, equalized, and scaled, providing a strong foundation for building hybrid deep learning models. Multi-axis sensor data (gyroscope and accelerometer), as shown in Fig. 1, were graphed during all activities involving the left ankle and right wrist. These time-series plots provided more insight into differences in the dynamics of various body parts' motions across different physical activities, thereby validating the usefulness of multi-sensor fusion for robust activity recognition.

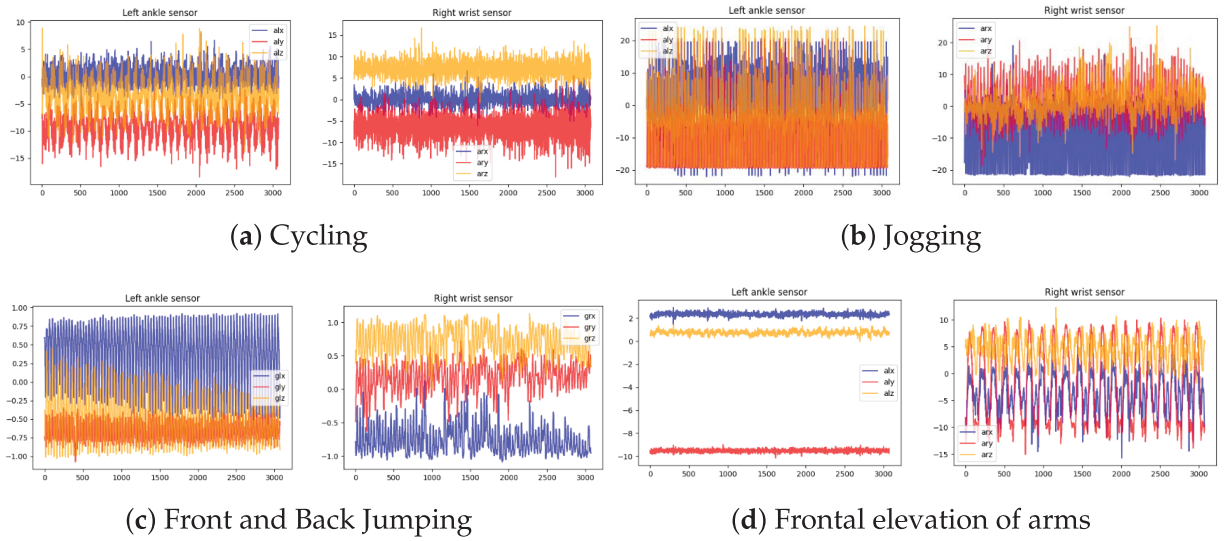


Figure 1: Time-series plots: sensor signals across activities. Each subplot visualizes accelerometer and gyroscope readings for different motion patterns, highlighting periodicity and intensity variations across activities

3.2 System Design

The proposed framework for HAR, shown in Fig. 2, involves several steps through which the design undergoes. Accelerometer and gyroscope sensors are placed on the left ankle and the right wrist, respectively, to record both lower-limb and upper-limb movements. The initial processing involves cleaning raw signals by removing outliers, assigning labels to signals, normalizing signals using robust scaling, and dividing the raw signals into training and testing sets to ensure that the input is delivered at the same quality. The processed data are then sent to the XGBoost classifier, which produces probability-based feature representations of activities via an ensemble of gradient-boosted decision trees. These probability characteristics are then fed into a Dense Neural Network (DNN), which performs deep feature learning via a series of dense layers with batch normalization and dropout regularization. Finally, a softmax layer is added. Lastly, the network makes predictions in 13 different activity categories, including both inactive and active states, as well as standing, sitting, walking, and running. It is a hybrid pipeline combining machine learning and deep learning, designed to classify human activities with high accuracy.

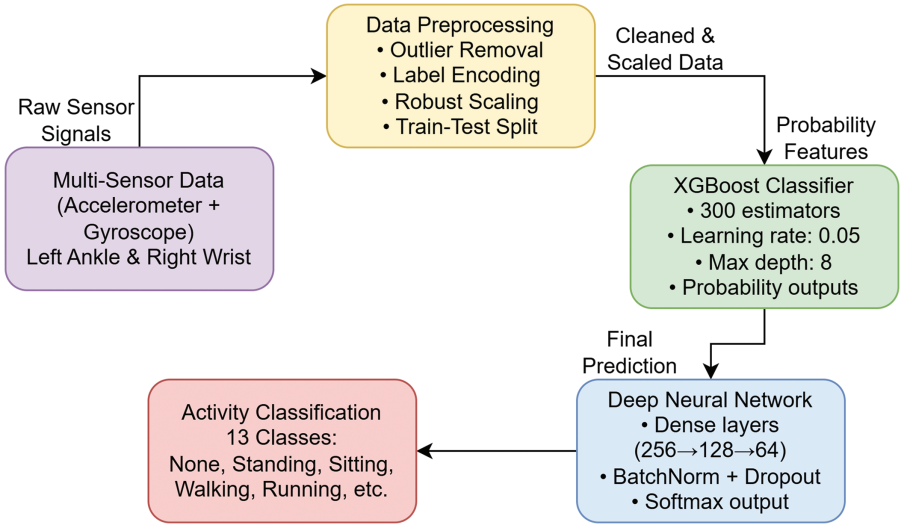


Figure 2: Basic system architecture: the IoT-enabled HAR system integrates sensor data acquisition, preprocessing, feature extraction, and hybrid ML-DNN classification

4 Proposed Approach

The proposed HAR model combines machine learning and deep learning to achieve higher classification and generalization. Fig. 3 illustrates a completed workflow that begins with raw sensor signals from the accelerometer and gyroscope modules and proceeds through an organized preprocessing pipeline that converts these signals into normalized, cleaned data. The pipeline follows the steps outlined in Algorithm 1, which involves extracting probability-based features using an Extreme Gradient Boosting (XGBoost) model and subsequently combining them with a Dense Neural Network (DNN) to perform final classification. This hybrid architecture leverages the benefits of interpretable and scalable feature-based ensemble algorithms, as well as the ability of deep neural networks to learn powerful feature representations, to enable effective recognition of human activities across a wide range of sensor modalities.

Rather than feeding the DNN with raw feature embeddings, the output of the machine learning (ML) models, in the form of probabilities, was used as input to the DNN to leverage the discriminative ability of both ensemble and non-ensemble learners. ML models can represent nonlinear interactions among features and provide class-level probabilities that reflect the high-level decision boundaries learnt during training. These probabilistic outputs are effectively task-specific, compact representations of the input data, helping reduce system noise and redundancy in the original feature space. By entering these class probabilities into the DNN, higher-order correlations can be learned and finer decision patterns refined, rather than re-learning lower-level feature interactions, resulting in a more stable and more efficient classification.

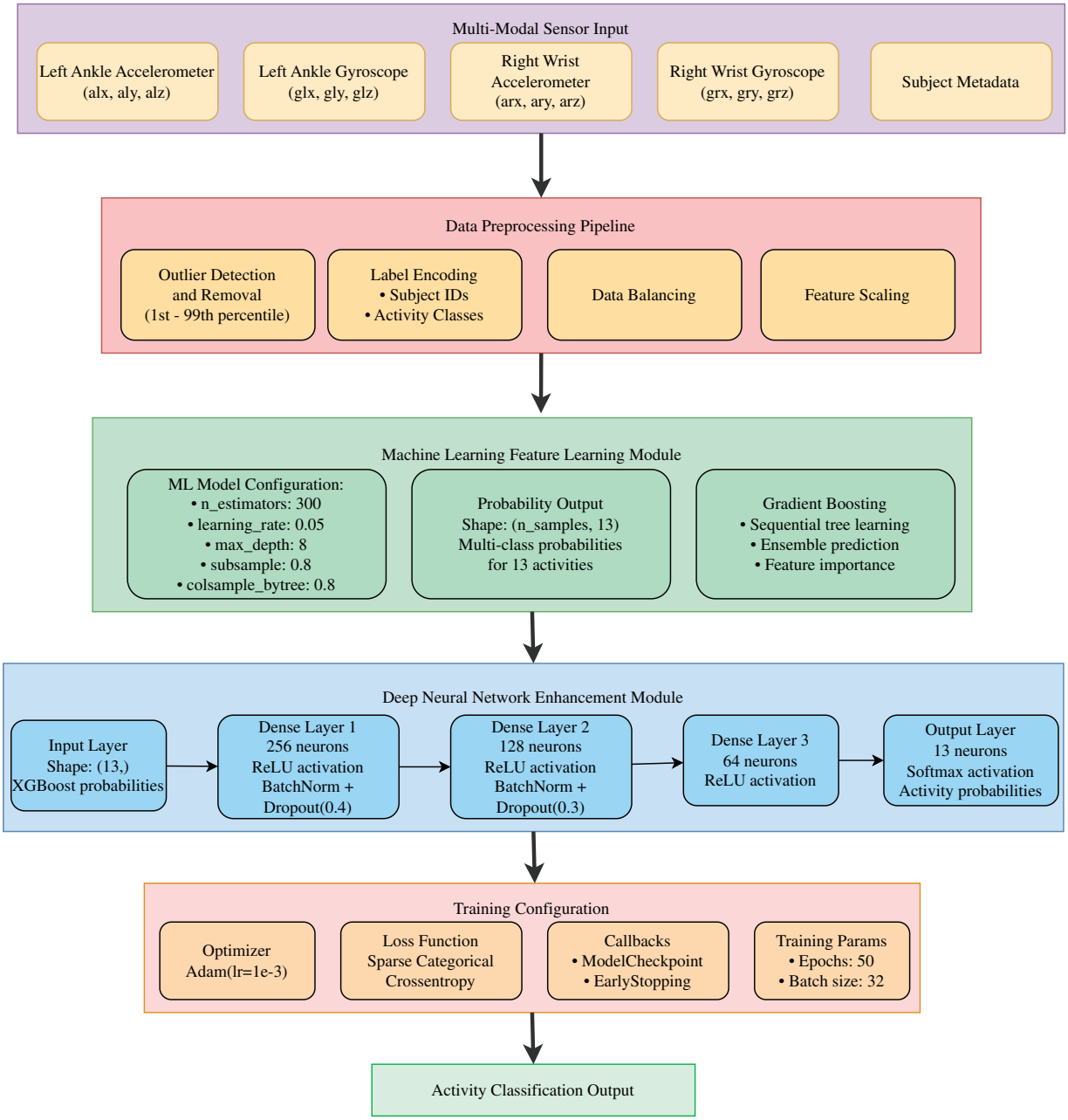


Figure 3: Detailed Framework for Hybrid ML-DNN-based Human Activity Recognition: This framework outlines the complete workflow, including sensor signal, Preprocessing, classical ML model training, probability generation, and final deep neural network-based refinement

Algorithm 1: Hybrid ML-DNN framework for human activity recognition

Require: Multi-sensor dataset $D = \{(x_i, y_i)\}_{i=1}^{N}$ where $x_i \in \mathbb{R}^{23}$ (sensor features) and $y_i \in \{0, 1, \ldots, 12\}$ (activity labels)

Ensure: Trained hybrid model for activity classification

1: **Phase 1: Data Preprocessing**

(Continued)

Algorithm 1 (continued)

2: Remove outliers using 1st-99th percentile filtering
3: Apply label encoding to subjects and activities
4: Balance dataset by downsampling the majority class (Activity 0) to 40,000 samples
5: Split data: (X_{train}, y_{train}), (X_{test}, y_{test}) with ratio 75:25
6: Apply RobustScaler: X_{train}^{scaled} = RobustScale(X_{train}), X_{test}^{scaled} = RobustScale(X_{test})
7: **Phase 2: ML Feature Learning**
8: Initialize ML classifier:
9: Train XGBoost: ML.$fit(X_{train}^{scaled}, y_{train})$
10: Extract probability features:
11: P_{train} = ML.$predict\_proba(X_{train}^{scaled}) \in \mathbb{R}^{N_{train} \times 13}$
12: P_{test} = ML.$predict\_proba(X_{test}^{scaled}) \in \mathbb{R}^{N_{test} \times 13}$
13: **Phase 3: Deep Neural Network Enhancement**
14: Define DNN architecture:
15: Input layer: Input(13)
16: Hidden layer 1: Dense(256) + ReLU + BatchNorm + Dropout(0.4)
17: Hidden layer 2: Dense(128) + ReLU + BatchNorm + Dropout(0.3)
18: Hidden layer 3: Dense(64) + ReLU
19: Output layer: Dense(13) + Softmax
20: Configure training parameters:
21: Optimizer: Adam with learning rate 1×10^{-3}
22: Loss function: Sparse Categorical Crossentropy
23: Callbacks: ModelCheckpoint, EarlyStopping (patience = 10)
24: **for** epoch = 1 to 50 **do**
25: DNN.$fit(P_{train}, y_{train}, batch\_size = 32)$
26: Perform Validation
27: **if** early stopping criteria met **then**
28: **break**
29: **end if**
30: **end for**
31: **Phase 4: Inference and Evaluation**
32: Predict activities: $\hat{Y}_{prob}$ = DNN.$predict(P_{test})$
33: Get final predictions: $\hat{Y}$ = arg max$(\hat{Y}_{prob})$
34: Evaluate performance: Accuracy, Precision, Recall, F1-Score
return Trained hybrid model (ML, DNN) and evaluation metrics

4.1 Machine Learning Models

This study examines several classical machine learning models for their interpretability, efficiency, and ability to handle complex data relationships. The Random Forest (RF) model combines predictions from multiple decision trees through ensemble learning, reducing overfitting by averaging outcomes or using majority voting. XGBoost constructs decision trees sequentially, with each tree correcting errors from previous trees, using a well-defined objective function to improve speed and performance. Gradient Boosting (GB) also builds trees sequentially, focusing on minimizing errors, but is generally more resource-intensive compared to Random Forest. K-Nearest Neighbors (KNN) classifies data points based on the majority label of the k nearest training samples, using distance metrics such as the Euclidean distance. However, it can

be computationally slow with larger datasets. Decision Trees recursively split data based on feature values using measures like the Gini index, offering clear interpretability; however, they are prone to overfitting if grown too complex, making them suitable as base models in ensemble methods. Overall, these models are compared for their performance and ease of interpretation in the context of the study.

4.2 Dense Neural Network (DNN)

One of the simplest possible deep learning architectures is the Dense Neural Network (also referred to as a fully connected neural network). They have many layers of connected neurons, so that a neuron in one layer is linked to all the neurons in the second layer. When applied to the HAR task in the IoT, DNNs can learn nonlinear relationships between raw or processed sensor features and activity categories. The rich structure enables the model to capture cross-sensor dependencies and temporal correlations among the input features. Eq. (6) demonstrates the forward propagation of a DNN layer mathematically.

$$z^{(l)} = W^{(l)} a^{(l-1)} + b^{(l)} \tag{6}$$

where $z^{(l)}$ is the linear combination at layer l, $W^{(l)}$ and $b^{(l)}$ denote the weights and bias parameters, and $a^{(l-1)}$ is the activation from the previous layer. A nonlinear activation function in Equation is then used to obtain the output activation as shown in Eq. (7).

$$a^{(l)} = \sigma(z^{(l)}) \tag{7}$$

where $\sigma(\cdot)$ can represent functions such as ReLU, sigmoid, or tanh, depending on the design choice. The last output layer can use the softmax activation to convert the logits into class probabilities, as provided in Eq. (8).

$$\hat{y}_i = \frac{\exp(z_i)}{\sum_{j=1}^{C} \exp(z_j)} \tag{8}$$

where C is the total number of activity classes, and $\hat{y}_i$ denotes the probability assigned to class i. The model is trained by minimizing a categorical cross-entropy loss function, defined in Eq. (9).

$$\mathcal{L} = -\sum_{i=1}^{C} y_i \log(\hat{y}_i) \tag{9}$$

where y_i is the true label (one-hot encoded) and $\hat{y}_i$ is the predicted probability from Eq. (9). In backpropagation, it computes the gradients of the loss function with respect to weights and biases by applying the chain rule. It uses them to update the parameters based on Eq. (10).

$$W^{(l)} \leftarrow W^{(l)} - \eta \frac{\partial \mathcal{L}}{\partial W^{(l)}} \tag{10}$$

where η denotes the learning rate, gradient descent enables the network to learn and progressively reduce classification error. DNNs are well-suited for HAR tasks due to their ability to model highly nonlinear feature interactions across heterogeneous IoT sensors. Yet they require substantial amounts of data to generalize effectively and can overfit without regularization methods, such as dropout or batch normalization.

4.3 K-Fold Cross Validation

To guarantee the thoroughness and external validity of the suggested hybrid learning model, the K-fold cross-validation was included in the assessment stage. Here, the dataset is divided into K equally sized folds, where one fold is used as the test set and the remaining K − 1 as the training set. The rotation will continue

until all folds are validated, and the final performance is calculated as the mean across the different runs. This approach will reduce the risk of biased estimates from a single train-test split and from performance reporting variance. It will offer a more valid assessment of model stability across varied data distributions. By applying K-fold cross-validation, the method will enable rigorous, consistent evaluation not only in the ML-only setting but also in the hybrid ML + DNN setting.

4.4 Hyperparameter Optimization

All machine learning models, including XGBoost (XGB), Random Forest (RF), Gradient Boosting (GB), K-Nearest Neighbors (KNN), and Decision Tree (DT), had their hyperparameters optimized iteratively by hand. In the ensemble-based models (XGB, RF, and GB), parameters such as the number of estimators, maximum tree depth, learning rate, and sub-sampling ratios were manually tuned based on validation set performance to balance bias and variance. Repetitive trials were used to optimize the number of neighbors and the distance metric in the KNN model to improve classification accuracy. In the DT model, the depth and splitting criteria were set to their maximum values to prevent overfitting and improve generalization. The manual tuning procedure was based on systematic observation of verification metrics (accuracy, F1-score, and loss trends) to select the most stable and effective configurations for each model.

5 Experimental Analysis and Results

To evaluate the proposed model's performance, several metrics are used. Accuracy, defined as $\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$, measures the percentage of correctly classified instances, where TP, TN, FP, and FN represent true positives, true negatives, false positives, and false negatives, respectively. However, accuracy may be inadequate in the presence of class imbalance, leading to the inclusion of precision and recall. Precision is defined as $\text{Precision} = \frac{TP}{TP+FP}$. It measures the reliability of positive predictions, while recall (or sensitivity) is given by $\text{Recall} = \frac{TP}{TP+FN}$, indicating the model's ability to detect positive instances. The F1-score, which balances precision and recall, is represented as $F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$.

5.1 Results

This subsection summarizes the experimental results of hybrid HAR models that combine machine learning classifiers with deep neural networks. A descriptive comparison of the five hybrid ML + DNN models tested with 3-fold cross-validation is presented in Table 3, which provides insight into the models' predictive power and stability as measured by the metrics. RF + DNN is clearly the most effective, with the highest accuracy, precision, recall, and F1-score across all folds, indicating its superior ability to learn the interactions among complex features through the combined power of ensemble learning and deep representations. XGB + DNN and KNN + DNN also provide competitive, stable results, with slight variations across folds, indicating that both gradient boosting and distance-based learning exhibit a more substantial effect of deep feature refinement. Conversely, GB + DNN and DT + DNN have relatively lower performance, but their scores are internally consistent, implying reliable but less articulate learning behavior. Overall, the performance of all models demonstrates the value of incorporating DNN components into the conventional ML framework, underscoring the relevance of hybrid modelling for achieving superior, reliable forecasts.

The RF+DNN model achieved 97% accuracy, with strong performance across most activity classes, maintaining precision and recall above 0.97, even for the minority class 4, as shown in Table 4. The second-best model, XGB + DNN, reached 97% accuracy and performed well across several activities, though it showed slight sensitivity to class imbalance in class 7. The GB + DNN hybrid scored 91%, underperforming the other models, particularly in classes 7 and 4, suggesting sensitivity to data imbalance. The KNN + DNN model achieved 96% accuracy, excelling at regular activities but showing limitations in class 7 due

to an uneven data distribution. Lastly, the DT + DNN hybrid achieved 92% accuracy, performing well on output classes with ample samples but struggling with minority classes, suggesting challenges with handling unbalanced datasets. Overall, RF + DNN and XGB + DNN emerged as the most effective models.

Table 3: K-fold performance comparison of ML + DNN models

Model	Metric	K-Fold 1	K-Fold 2	K-Fold 3	Average
RF + DNN	Accuracy	0.9712	0.9691	0.9707	0.9703
	Precision	0.9685	0.9702	0.9696	0.9694
	Recall	0.9719	0.9694	0.9714	0.9709
	F1 Score	0.9691	0.9703	0.9698	0.9697
XGB + DNN	Accuracy	0.9541	0.9526	0.9539	0.9535
	Precision	0.9507	0.9523	0.9514	0.9515
	Recall	0.9558	0.9561	0.9589	0.9569
	F1 Score	0.9534	0.9528	0.9542	0.9535
GB + DNN	Accuracy	0.9108	0.9119	0.9122	0.9116
	Precision	0.9061	0.9078	0.9074	0.9071
	Recall	0.9107	0.9124	0.9145	0.9125
	F1 Score	0.9068	0.9083	0.9087	0.9079
KNN + DNN	Accuracy	0.9562	0.9538	0.9543	0.9548
	Precision	0.9515	0.9528	0.9523	0.9522
	Recall	0.9557	0.9545	0.9560	0.9554
	F1 Score	0.9533	0.9526	0.9530	0.9530
DT + DNN	Accuracy	0.9175	0.9154	0.9176	0.9168
	Precision	0.9142	0.9156	0.9157	0.9151
	Recall	0.9149	0.9163	0.9164	0.9159
	F1 Score	0.9151	0.9150	0.9160	0.9154

Table 4: Comparison of classification performance of hybrid models (RF + DNN, XGB + DNN, GB + DNN, KNN + DNN, DT + DNN)

Model	Metric	0	1	2	3	4	5	6	7	8	9	10	11	12	Acc.
RF+DNN	Prec.	0.96	0.98	0.99	0.95	0.95	0.97	0.99	0.94	0.96	0.99	0.98	0.98	0.97	
	Rec.	0.98	0.99	1.00	0.98	0.91	0.99	1.00	0.83	0.97	1.00	1.00	1.00	0.99	0.97
	F1	0.97	0.99	0.99	0.97	0.93	0.98	0.99	0.88	0.97	0.99	0.99	0.99	0.98	
XGB+DNN	Prec.	0.94	0.98	0.99	0.96	0.93	0.97	0.99	0.93	0.96	0.98	0.98	0.98	0.96	
	Rec.	0.98	0.99	0.99	0.98	0.93	0.99	1.00	0.80	0.97	1.00	0.99	0.99	0.99	0.97
	F1	0.96	0.99	0.99	0.97	0.93	0.98	0.99	0.86	0.97	0.99	0.99	0.98	0.97	
GB+DNN	Prec.	0.84	0.96	0.96	0.88	0.82	0.91	0.99	0.79	0.87	0.98	0.97	0.92	0.88	
	Rec.	0.90	0.98	0.98	0.90	0.76	0.92	0.99	0.59	0.93	1.00	0.99	0.96	0.94	0.91
	F1	0.87	0.97	0.97	0.89	0.79	0.91	0.99	0.68	0.90	0.99	0.98	0.94	0.91	

(Continued)

Table 4 (continued)

Model	Metric	0	1	2	3	4	5	6	7	8	9	10	11	12	Acc.
KNN+DNN	Prec.	0.96	0.97	0.96	0.93	0.90	0.96	0.99	0.93	0.95	0.98	0.98	0.97	0.95	0.96
	Rec.	0.98	1.00	0.99	0.96	0.88	0.99	1.00	0.76	0.95	1.00	1.00	0.99	0.99	
	F1	0.97	0.98	0.98	0.95	0.88	0.97	0.99	0.84	0.95	0.99	0.99	0.98	0.97	
DT+DNN	Prec.	0.90	0.97	0.96	0.91	0.81	0.92	0.99	0.73	0.91	0.98	0.98	0.95	0.93	0.92
	Rec.	0.91	0.97	0.97	0.91	0.79	0.95	0.99	0.71	0.91	0.98	0.98	0.96	0.93	
	F1	0.90	0.97	0.97	0.91	0.80	0.94	0.99	0.72	0.91	0.98	0.98	0.95	0.93	

Table 5 provides a comparison of the ablation study of the models. RF + DNN was the most successful model, followed closely by XGB + DNN and KNN + DNN. GB + DNN and DT + DNN had lower accuracies, suggesting they did not integrate well with deep learning and did not capture the complexities of HAR. The results together underscore that ensemble-based approaches (RF and XGB) in conjunction with DNNs offer better generalization and balanced recognition across all activity classes.

Table 5: Ablation study comparing ML-only, DNN-only, and Hybrid configurations for HAR

Category	Model	Accuracy	Precision	Recall	F1 Score
ML-only	RF	0.9718	0.9706	0.9725	0.9709
	XGB	0.9532	0.9521	0.9553	0.9525
	GB	0.9001	0.9006	0.9001	0.9001
	KNN	0.9388	0.9366	0.9358	0.9327
	DT	0.9216	0.9197	0.9202	0.9199
DNN-only	DNN	0.8827	0.8831	0.8772	0.8700
Hybrid (ML + DNN)	RF + DNN	0.9703	0.9694	0.9709	0.9697
	XGB + DNN	0.9535	0.9515	0.9569	0.9535
	GB + DNN	0.9116	0.9071	0.9125	0.9079
	KNN + DNN	0.9548	0.9522	0.9554	0.9530
	DT + DNN	0.9168	0.9151	0.9159	0.9154

A multifaceted ablation study of HAR across ML-only, DNN-only, and Hybrid (ML + DNN) setups is presented in Table 5, demonstrating distinct performance trends across the model categories. Random Forest has the highest accuracy and F1-score among the ML-only models; next in line are XGBoost and KNN, and the performance of Gradient Boosting and Decision Tree is relatively low. The individual DNN model achieves significantly inferior results compared to most ML methods, suggesting that deep learning alone may not fully capture the discriminative structure of this feature space. Nevertheless, in hybrid configurations in which ML models are embedded in representations based on DNN, performance becomes more stable and, in some cases, better. RF + DNN and XGB + DNN seem to achieve good, competitive results, demonstrating the usefulness of combining deep feature extraction with classical ML decision boundaries. Even previously weaker models, including GB and DT, demonstrate improved stability and a minor increase in recall and F1-scores when trained in combination with DNNs. In general, the findings of the ablation studies indicate that although ML models tend to be stronger than DNN models, the hybrid model offers

a moderate, and in many cases superior, approach, combining the respective strengths to generate more resilient and generalized HAR predictions.

Fig. 4 shows the training and validation accuracy curves of the five hybrid models. Fig. 4a RF + DNN model has close to perfect training accuracy, but the difference between training and validation accuracy indicates slight overfitting. Equally, the XGB + DNN in Fig. 4b and GB + DNN in Fig. 4c models show a gradual increase in accuracy as the epoch progresses, albeit that the validation accuracy falls short of the training accuracy. Fig. 4d shows that there is no significant variation in the accuracy of the KNN + DNN model, thus demonstrating its stability. Finally, in Fig. 4e, the DT + DNN model is highly accurate at the beginning and maintains that performance throughout the epochs. In general, the curves indicate that all models can learn, but some have a better generalizing ability than others.

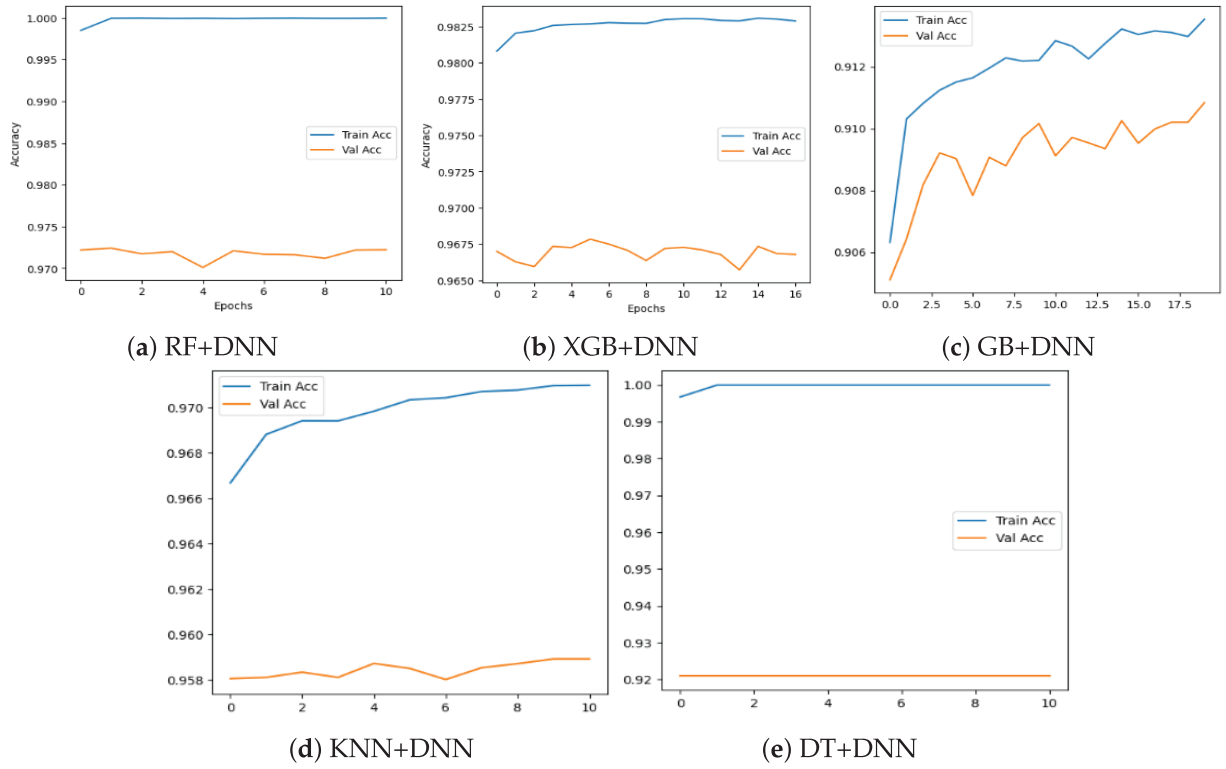
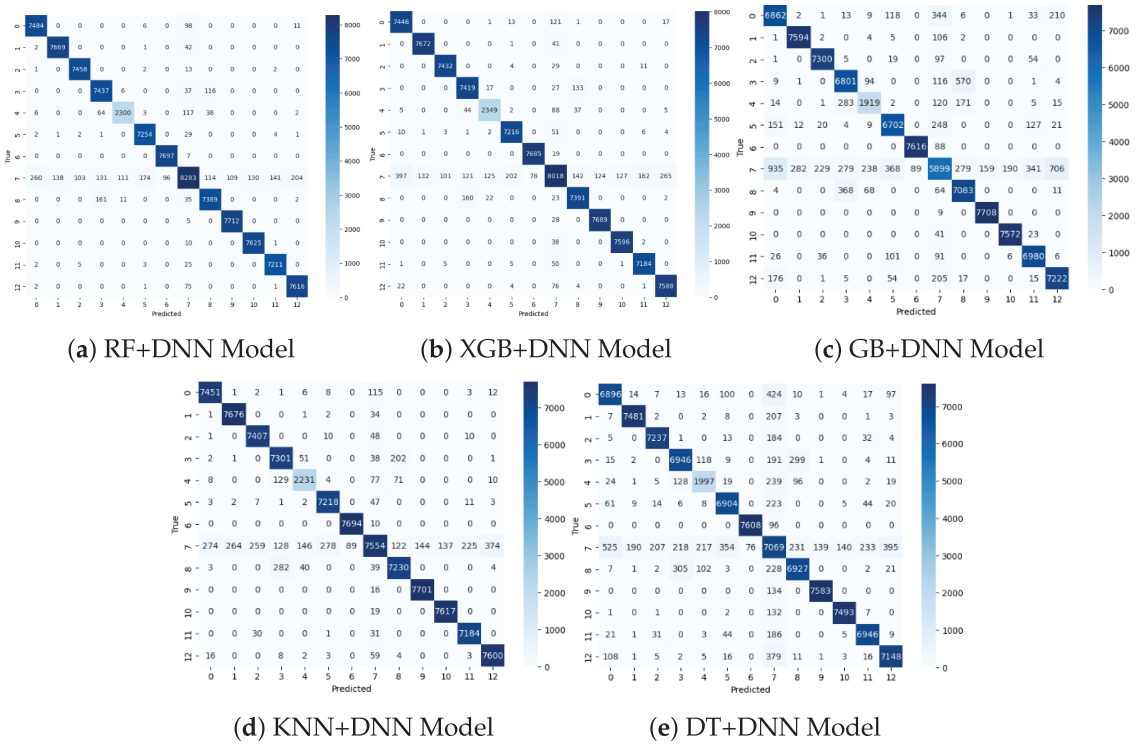


Figure 4: Training and validation accuracy curves of the proposed hybrid Machine Learning–Deep Neural Network (ML–DNN) models

Fig. 5 presents the confusion matrices for each hybrid model across 12 activity classes. RF + DNN in Fig. 5a shows very sharp diagonals, indicating pronounced classification of most of the activities. The XGB + DNN in Fig. 5b does the same thing, except there are a few misclassifications that are scattered to the neighboring classes. The GB + DNN of Fig. 5c matrix exhibits a minimal bit of confusion in the overlapping activities, which is why its performance is relatively lower than that of RF and XGB. The KNN + DNN model in Fig. 5d achieves high accuracy but exhibits a couple of systematic confusions in the mid-range classes only. Lastly, the DT + DNN in Fig. 5e delivers credible performance, but with significantly weaker performance than RF and XGB due to greater off-diagonal errors. Overall, these matrices indicate that RF + DNN and XGB + DNN consistently yield the cleanest activity matrices.

In summary, the aggregate findings of the ablation table and the analysis with comparative accuracy plot Fig. 6 evidently indicate the advantages of the combination of deep learning with classical machine

learning tools in Human Activity Recognition. Although ML-only models, especially RF and XGB, offer excellent baseline performance, hybrid ML + DNN configurations provide a more balanced and robust alternative that, by far, beats the DNN-only configuration and compares favorably with the best ML baselines. The hybrid models successfully leverage DNN-based feature representations while preserving the structured decision-making advantages of conventional ML classifiers. The results in more stable metrics and improved generalization, demonstrating that the suggested hybrid structure is a more successful and stable solution to HAR than either ML or DNN networks alone.



(**a**) RF+DNN Model (**b**) XGB+DNN Model (**c**) GB+DNN Model

(**d**) KNN+DNN Model (**e**) DT+DNN Model

Figure 5: Confusion matrices of the proposed hybrid Machine Learning–Deep Neural Network (ML–DNN) models

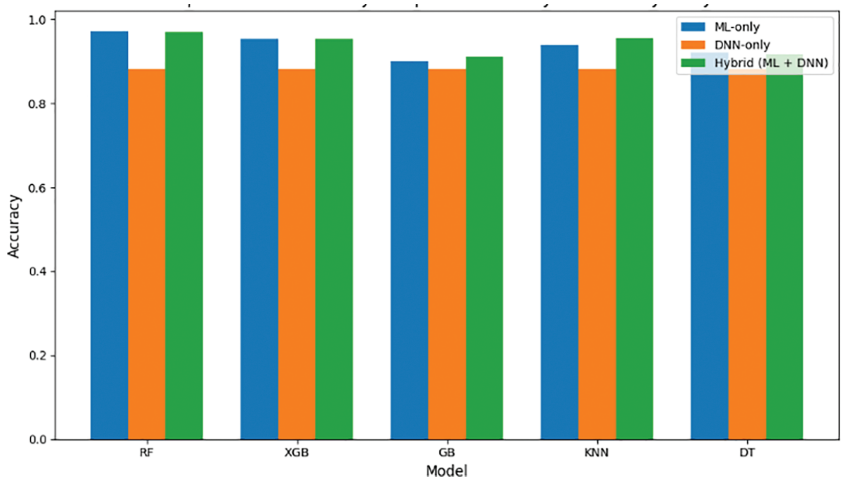


Figure 6: Accuracy comparison showing Hybrid (ML + DNN) outperforming DNN-only and closely matching strong ML baselines

5.2 Discussion

A comparative study has been conducted between the proposed hybrid ML + DNN models and a recent research [17] on wearable sensor-based assistive technologies, as shown in Table 6. The results demonstrate that our proposed method achieves superior performance. The substantial improvement achieved by our RF + DNN approach can be attributed to its hybrid architecture, which effectively combines the feature selection capabilities of RF with the deep representation learning of neural networks.

Table 6: Comparative performance analysis with state-of-the-art methods

Architecture	Dataset	Accuracy
MLP	mHealth Dataset	90.55%
XGBoost	mHealth Dataset	89.97%
CNN	mHealth Dataset	83.91%
ConvLSTM	mHealth Dataset	83.89%
AE w/RF	mHealth Dataset	83.27%
LSTM	mHealth Dataset	78.09%
RF+DNN (Proposed)	**mHealth Dataset**	**97.03%**

Compared to the base study, which reported much greater computational needs (7632, 4632 s, 4421, and 3611 MB for the mHealth and ScientISST MOVE datasets, respectively), our hybrid machine learning and deep learning models exhibited much better computational efficiency. The Decision Tree + DNN model took the shortest 161.68 s to train, followed by KNN+DNN (218.26 s), Random Forest + DNN (345.85 s), Gradient Boosting+DNN (556.29 s), and XGBoost + DNN (1265.85 s). Although training time varies across these configurations as well, all hybrid models trained successfully in a few minutes, resulting in a significant reduction in computational efficiency compared to the base approach. Such efficiency is what makes the proposed framework better suited to real-world, resource-limited contexts, including mobile and wearable healthcare systems, where it is crucial to achieve reduced computation time and lower memory consumption to enable continuous, scalable activity recognition.

5.3 Computational Performance Evaluation

Every hybrid model was trained on a CPU-based system with an x86_64 architecture (4 cores) and 31.35 GB of RAM, without dedicated GPU acceleration. TensorFlow was used with the CPU device for all deep learning operations to ensure consistent hardware conditions across experiments. The computational time was very different based on the underlying machine learning model incorporated with the DNN. The XGB + DNN hybrid had the longest training time, about 1265.85 s (21.10 min). This long time can be explained by the fact that the XGBoost boosting algorithm is iterative and additive, i.e., weak learners are optimized sequentially. In contrast, the DNN fine-tuning strategy is based on backpropagation. The GB+DNN model also had a relatively high computational cost (556.29 s; 9.27 min) due to sequential tree construction and gradient-based optimization, although it was more cost-effective than XGBoost because of its simpler ensemble configuration. The RF+DNN hybrid had an acceptable compromise between computational and performance, taking 345.85 s (5.76 min) to train. RF: The parallelizability of random forests, as an ensemble methodology, enables independent tree training, reducing the computational cost of boosting-based algorithms without compromising random forest performance. The non-parametric KNN + DNN model, with 218.26 s (3.64 min) of training time, further reduced computational requirements by eliminating the need for complex iterative KNN training before the DNN step. The DT + DNN model took

the least time to train, at 161.68 s (2.69 min). This is done because a single decision tree requires fewer recursive partitions than ensemble methods.

In general, the findings show that boosting-based hybrids such as XGB + DNN and GB + DNN have strong learning ability due to their iterative refinement mechanism, but incur significantly higher computational costs. Simpler hybrids like DT + DNN and KNN + DNN, in contrast, have fast training times but use fewer resources and can be deployed in real time or on resource-constrained IoT systems. The RF + DNN model is balanced; it has high predictive accuracy (97.03%) and a moderately long training time; hence, it should be the preferred tradeoff between computational complexity and model performance.

6 Conclusion and Future Scope

This study proposes a hybrid HAR framework that combines traditional machine learning and DNN to leverage both handcrafted statistical features and the deep feature representations of wearable sensor data. The two paradigms together provided a more comprehensive feature learning process, strengthened classification, and facilitated generalization. RF + DNN has the highest performance with the accuracy of 97.03%, precision of 97.12%, recall of 97.25% and F1-score of 97.15%, and therefore it is better than the other combinations of XGB + DNN, GB + DNN, KNN + DNN and DT + DNN. The visual analysis, using confusion and ROC curves, also showed that RF + DNN and XGB + DNN are beneficial, achieving greater separation of activity classes and fewer misclassifications across a broad range of human behaviors. These findings suggest that the relationship between ensemble learning and deep neural networks can be leveraged to achieve a balanced tradeoff between interpretability, feature diversity, and learning capacity. Attention-based graph neural networks and transformer architectures can be utilized in the future to enhance implementations, particularly by considering temporal pattern recognition. Moreover, real-time HAR can be implemented and deployed to lightweight hybrid models running on edge devices and wearables, making the framework more relevant in other fields, such as healthcare monitoring, fitness tracking, and assistive systems that utilize smart IoT.

Acknowledgement: We acknowledge the support via funding from Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2026R909), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Funding Statement: This research has been supported by Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2026R909), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Author Contributions: The authors confirm their contributions to the paper as follows: study conception and design: Shtwai Alsubai, Abdullah Al Hejaili, Vincent Karovič; data collection: Shtwai Alsubai, Abdullah Al Hejaili; Analysis and interpretation of results: Shtwai Alsubai, Abdullah Al Hejaili, Najib Ben Aoun, Amina Salhi, Vincent Karovič. Draft manuscript preparation: Shtwai Alsubai, Abdullah Al Hejaili, Najib Ben Aoun, Amina Salhi, Vincent Karovič. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are openly available in the UCI Repository at [https://archive.ics.uci.edu/dataset/319/mhealth+dataset].

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Abdulhussain SH, Mahmmod BM, Alwhelat A, Shehada D, Shihab ZI, Mohammed HJ, et al. A comprehensive review of sensor technologies in IoT: technical aspects, challenges, and future directions. Computers. 2025;14(8):342. doi:10.3390/computers14080342.
2. Malik S, Rana A. The rise of smart technologies: empowering the IoT revolution in everyday life. AN-WESH Int J Manag Inf Technol. 2025;10(1):34.
3. Hu L, Shu Y. Enhancing decision-making with data science in the internet of things environments. Int J Adv Comput Sci Appl. 2023;14(9):1151–62. doi:10.14569/ijacsa.2023.01409120.
4. Issa ME, Helmi AM, Al-Qaness MAA, Dahou A, Abd Elaziz M, Damaševičius R. Human activity recognition based on embedded sensor data fusion for the internet of healthcare things. Healthcare. 2022;10(6):1084. doi:10.3390/healthcare10061084.
5. Ni J, Tang H, Haque ST, Yan Y, Ngu AH. A survey on multimodal wearable sensor-based human action recognition. arXiv:240415349. 2024.
6. Kumar P, Chauhan S, Awasthi LK. Human activity recognition (har) using deep learning: review, methodologies, progress and future research directions. Arch Comput Meth Eng. 2024;31(1):179–219. doi:10.1007/s11831-023-09986-x.
7. Ul Haq M, Sethi MAJ, Ben Aoun N, Alluhaidan AS, Ahmad S, Farid Z. CapsNet-FR: capsule networks for improved recognition of facial features. Comput Mater Continua. 2024;79(2):2169–86. doi:10.32604/cmc.2024.049645.
8. Mzoughui MC, Ben Aoun N, Naouali S. A review on kinship verification from facial information. Vis Comput. 2024;41(3):1789–809. doi:10.1007/s00371-024-03493-1.
9. Al Mudawi N, Azmat U, Alazeb A, Alhasson HF, Alabdullah B, Rahman H, et al. IoT powered RNN for improved human activity recognition with enhanced localization and classification. Sci Rep. 2025;15(1):10328. doi:10.1038/s41598-025-94689-5.
10. Karim M, Khalid S, Lee S, Almutairi S, Namoun A, Abohashrh M. Next generation human action recognition: a comprehensive review of state-of-the-art signal processing techniques. IEEE Access. 2025;13:135609–33. doi:10.1109/access.2025.3590073.
11. Khan MZ, Usman M, Ahmad J, Rahman MMU, Abbas H, Imran M, et al. Tag-free indoor fall detection using transformer network encoder and data fusion. Sci Rep. 2024;14(1):16763. doi:10.1038/s41598-024-67439-2.
12. Hussain A, Khan SU, Khan N, Bhatt MW, Farouk A, Bhola J, et al. A hybrid transformer framework for efficient activity recognition using consumer electronics. IEEE Trans Consumer Electron. 2024;70(4):6800–7. doi:10.1109/tce.2024.3373824.
13. Verma N, Mundody S, Guddeti RMR. An efficient AI and IoT enabled system for human activity monitoring and fall detection. In: Proceedings of the 2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT); 2024 Jun 24–28; Kamand, India. p. 1–6.
14. Aziz A, Mirzaliev S, Maqsudjon Y. Real-time monitoring of activity recognition in smart homes: an intelligent IoT framework. J Intell Syst Internet Things. 2023;10(1):76–83. doi:10.54216/jisiot.100106.
15. Lakoju M, Ajienka N, Khanesar MA, Burnap P, Branson DT. Unsupervised learning for product use activity recognition: an exploratory study of a "chatty device". Sensors. 2021;21(15):4991. doi:10.3390/s21154991.
16. Janaki M, Balakrishnan S. Leveraging ensemble learning models for human activity recognition. Indones J Electr Eng Inform. 2025;13(1):249–60. doi:10.52549/ijeei.v13i1.6140.
17. O'Halloran J, Curry E. A comparison of deep learning models in human activity recognition and behavioural prediction on the MHEALTH dataset. AICS. 2019;2563:212–23.