Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 10, No. 1, 1119-1137
2026
Publisher: Learning Gate
DOI: 10.55214/2576-8484.v10i1.11864

An intelligent approach for human activity analysis using data mining techniques

Ahmed Elkhatib[1], Ahmed Aboelfetouh[1,2], Hazem M. Elbakry[1*]
[1]Department of Information Systems, Faculty of Computers and Information, Mansoura University, Mansoura, Egypt; elkhateeb1983@mans.edu.eg, ahmed.elkhateeb1983@gmail.com (A.E.) elbakry@mans.edu.eg (H.M.E.).
[2]Delta Higher Institute for Management and Accounting Information Systems, Mansoura, Egypt; elfetouh@mans.edu.eg (A.A.)

Abstract: An intelligent information system for sensor-based human activity analysis using data mining techniques presents a comprehensive study on developing a system that employs data mining to analyze human activities based on sensor data. With advancements in wearable technologies and embedded sensor systems, such as smartphones, smartwatches, and various environmental and object-attached sensors, it is now possible to automatically and continuously recognize and track human activities through sensor data collection. These techniques are generally known as sensor-based human activity analysis and can be applied across multiple fields, including healthcare, entertainment, and artificial intelligence system design. The core approach involves abstracting sensor data into higher-level activity recognition through various data processing and mining methods. In this study, eight classifiers are applied to the HARSense dataset, including naïve Bayes (NB), decision tree, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), neural network, XGBoost, random forest (RF), and extra trees classifier (ETC). The models are evaluated on the HARSense dataset, with the extra trees classifier achieving the highest accuracy of 97.12%.

Keywords: *Data mining techniques, Extra trees classifier (ETC), HARSense dataset, Decision tree, Human activity recognition (HAR), Random forest (RF).*

1. Introduction

Beyond typical application areas where research in human activity analysis is often focused, such as health, sports, time and workplace management, or monitoring the qualitative behavior of patients with certain neurological impairments, smart environments frequently address activity analysis in situations where only raw sensor signals are available. These applications require sensory devices capable of recording signals over time related to human activities, which intelligent algorithms can recognize from these data sets. Experiences with this type of problem, even before the IoT revolution, led to the development of a sensor called HARSense. This sensor, through its components and a Data Mining tool, can generate the data necessary to solve human activity analysis problems. It has proven particularly useful in the simulation of realistic sensor data files [1-3].

Human activity recognition (HAR), in HARSense, involves using a computational model to automatically identify and classify human activities based on observations from sensors attached to the body, such as accelerometers, gyroscopes, magnetometers, or others. These small devices can accompany humans everywhere, unlike environmental sensors. In recent years, many researchers have focused on developing hardware and software to support various applications within smart environments [4, 5].

However, existing methods and applications of sensor-based human activity analysis have not fully explored the capability of data mining in processing raw sensor data and extracting more complex

History: Received: 12 November 2025; Revised: 5 January 2026; Accepted: 8 January 2026; Published: 23 January 2026
* Correspondence: elbakry@mans.edu.eg

activity features. At the same time, as a form of data-driven science, the emerging technology called "Big Data" heavily relies on data mining techniques for pattern analysis and knowledge discovery. In recent years, integrating data mining techniques into human activity analysis has gained increasing attention in the fields of activity monitoring and pervasive computing. It is important to study the use of modern data mining tools and methods and to develop new intelligent information systems for sensor-based human activity analysis. The outcomes of such studies will not only provide advanced solutions for current activity analysis approaches but also open opportunities for developing more sophisticated analyses and applications using recognized activity data [6].

Human action analysis has gained significant attention in computer vision and artificial intelligence over recent years. It is a crucial technology for context-aware systems and devices, providing essential inputs for understanding human intentions and behaviors in various settings. Typically, human activity recognition systems rely on sensory data collected by specialized sensors that monitor specific activities and environmental factors. These sensors generally include cameras, microphones, and accelerometers, which capture information about a subject's actions and movements across visual, auditory, and motor domains [7, 8].

The term of human activity within the context of the HARSense dataset refers to the process of humans making observable and actionable decisions in both real and virtual environments. To describe group behavior, unique patterns in motion and posture during various routine activities are identified. The collective responses observed are correlated with these activities. Support for understanding human activities is best achieved through analyzing human decision-making processes using learning methods. The research aims to perform this analysis from device logs employing machine learning and data mining techniques [1].

This paper employs machine learning and data mining techniques for human activity analysis in the HARSense dataset, utilizing eight classifiers: naive Bayes (NB), decision tree, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), neural network, XGBoost, random forest (RF), and extra trees classifier (ETC). We achieved promising results with the extra trees classifier.

The remainder of this study is organized as follows. Section 2 presents an overview of the most relevant studies and existing approaches related to activity recognition, highlighting their methodologies and limitations. Section 3 introduces the main machine learning classifiers employed in this work, outlining their theoretical foundations and comparative characteristics. Section 4 describes the proposed methodology, including dataset description, preprocessing steps, and evaluation measures. Section 5 reports and analyzes the experimental results obtained from both individual and ensemble classifiers, supported by performance metrics and confusion matrix visualizations. Finally, Section 6 concludes the study by summarizing key findings and discussing potential directions for future research.

2. Related Work

Hassan et al. [2] introduced a robust human action analysis framework utilizing deep learning and smartphone sensors. The method involved extracting efficient features from raw data, including autoregressive coefficients, median, and mean. These features were processed using kernel principal component analysis (KPCA) and linear discriminant analysis (LDA) to enhance robustness. Subsequently, a Deep Belief Network (DBN) was trained for effective action recognition. The approach was tested on twelve different physical actions, achieving a mean recognition rate of 89.61% and an overall accuracy of 95.85% [2].

Chin et al. [9] investigated Daily Actions analysis on the human actions primitives recognition dataset. They studied human action data captured through classification by a wrist-worn accelerometer. The classification was based on various daily activities performed by a normal person. A wrist-worn tri-axial accelerometer was used to collect acceleration data along the X, Y, and Z axes during each test. Nine statistical parameters, combined with energy spectral density and the relationships between accelerometer interpretations, were used to extract sixty-three features from the raw data. For the classification process, Ranker, Tabu Search, and Particle Swarm Optimization were employed to test and select the most relevant features. Classification algorithms such as Random Forest, Support Vector Machine, and k-Nearest Neighbors were implemented. The results showed that SVM with a radial basis function kernel achieved the highest accuracy, with a correct classification rate of 91.5% [9].

Padmaja et al. [10] introduced a random split point procedure using the Extra Trees method for human activity recognition. It generates K random split points from all dataset features and selects the best based on the maximum information gain score. The approach was tested on two datasets: HAR and HAPT, containing six and twelve activities, respectively. The HAR dataset includes smartphone sensor signals for three static and dynamic daily activities, while the HAPT dataset features six postural transitions from these activities. The proposed method achieved an accuracy of 92.63% on HAPT and 94.16% on HAR datasets, demonstrating its effectiveness in activity recognition tasks [10].

Bukht et al. [11] introduced a framework based on a decision tree for human activity recognition using feature fusion. The study emphasizes feature fusion and optimal feature reduction. The proposed method involves four main steps: first, frame preprocessing to enhance video contrast and remove noise; second, applying a statistical method for silhouette extraction; third, feature extraction and fusion using SIFT and ORB; and fourth, feature reduction with t-distributed stochastic neighbor embedding (t-SNE). The final step is action recognition via a decision tree. Experiments conducted on UT Interaction data achieved a recognition rate of 95% [11].

Khan et al. [12] developed a human action recognition model utilizing the Human Action Recognition Trondheim (HARTH) dataset. The model aims to identify various daily human activities in free-living environments, which are challenging due to unplanned actions. While controlled data can yield optimal results, real-world applications often face difficulties. The framework employs machine learning classifiers with time-domain features extracted from sensor data. Specifically, the multilayer perceptron (MLP) classifier achieved an accuracy of 92.92% [12].

Khan et al. [13] applied human action recognition in the wild, which involved selecting an in-the-wild, extra-sensory dataset comprising six activities: bicycling, walking, running, standing, sitting, and lying down. Three machine learning classifiers, decision trees, random forest, and k-nearest neighbors, were used for time domain feature extraction and human action recognition. The proposed system achieved an accuracy of 89.98% with the random forest classifier [13].

Zhu et al. [14] proposed applying the Extra-Trees classifier for human action recognition (HAR) using a wearable sensor device carried by the experimenter to collect motion data. The experiment compared three classifiers: decision tree, KNN, and extra-trees. The accuracy results were 87.75%, 90.77%, and 93.25%, respectively, with the extra-trees classifier achieving the highest accuracy [14].

Nematallah and Rajan [15] introduced a study that conducted a quantitative analysis of mother wavelet function selection for wearable sensors used in human action analysis. It employed a method combining wavelet packet transform with the energy-to-Shannon-entropy ratio, utilizing two classification algorithms: decision tree (DT) and support vector machines (SVM). The researchers examined six different mother wavelet families with varying numbers of vanishing points. Experiments were performed on eight datasets: HAR70+, REALDISP, PAMAP2, DaLiAc, HARsense, HARTH, WISDM Activity Prediction, and MHEALTH. The balanced accuracy achieved with decision tree (DT) and support vector machines (SVM) was 74.62% and 76.53%, respectively, using the Coif14-based wavelet packet transform [15].

Table 1 summarizes the related work representing the study, the dataset used, the applied method, and its performance measures.

ISSN: 2576-8484
Vol. 10, No. 1: 1119-1137, 2026
DOI: 10.55214/2576-8484.v10i1.11864

Table 1.
Related work summary.

Study	Dataset	Method	Advantages	Disadvantages	Application	Accuracy
Hassan, et al. [2]	Smartphone Dataset	DBN	Robust feature extraction using KPCA and DBN enables better activity distinction	Complex feature-engineering and training increase system difficulty	Fitness and Lifestyle Apps	95.85%
Chin, et al. [9]	Human Motion Primitives Detection Dataset	SVM	Utilizes a diverse age group (19–91 years) and multiple daily activities for broader relevance	Potential for reduced accuracy with specific activities due to wrist sensor placement	Wearable Activity Trackers	91.5%
Padmaja, et al. [10]	HAR HAPT	ETC	Reduced computational time and faster model building due to randomized split selection	Still requires a sizable dataset and careful parameter selection for optimal performance	Elderly Care	92.63% 94.16%
Bukht, et al. [11]	UT interaction data	DT	The method is simple, interpretable, and efficient using a decision tree.	Only one classifier (DT) is used, without comparison to more advanced models.	Surveillance systems	95%
Khan, et al. [12]	HARTH	MLP	Focuses on free-living environments, making the model more realistic than scripted setups.	Limited generalization since performance may differ across varied real-life conditions.	rehabilitation	92.92%
Khan, et al. [13]	in-the-wild extra-sensory dataset	RF	Evaluated on an extra-sensory dataset, enhancing real-world applicability.	Limited to only six predefined activities, restricting generalizability.	patient activity tracking	89.98%
Zhu, et al. [14]	HAR	ETC	Uses wearable sensor data, which is reliable and widely applicable.	Focuses only on traditional ML classifiers	motion-based interaction	93.25%
Nematallah and Rajan [15]	HARsense	DT SVM	Proposes an optimal mother wavelet selection method, improving HAR performance.	The computational complexity of the wavelet packet transform may hinder real-time applications.	activity-aware automation	74.62% 76.53%

3. Preliminaries

This section offers an overview of the machine learning classifiers used in this study. Each classifier has unique characteristics, assumptions, and learning strategies that affect its performance across different data domains. The classifiers include traditional statistical models, distance-based algorithms, tree-based ensembles, and neural architectures. Their theoretical foundations, advantages, and limitations are briefly summarized to provide context for their selection and comparative analysis within the proposed framework.

i. Naive Bayes Classifier (NB)
   Naive Bayes classifiers are a family of linear probabilistic classifiers assuming features are tentatively independent given the target class. The strength of this assumption is what gives the classifier its name. These classifiers are among the simplest Bayesian network models [16].
ii. Support Vector Machine (SVM) Classifier
   Support vector machines (SVMs) are supervised max-margin models with associated learning algorithms that analyze data for classification and regression analysis. SVMs can efficiently perform non-linear classification using the kernel trick [17].
iii. Decision Tree Classifier

A Decision Tree Classifier is a supervised machine learning algorithm that uses a tree-like model to classify data. It is a popular and widely used algorithm because of its effectiveness, interpretability, and simplicity. Decision Trees can handle missing values by using surrogate splits. They are robust to noisy data, as the tree structure can absorb some noise levels. However, Decision Trees may suffer from overfitting, especially when the tree is deep or the sample size is small. Additionally, training Decision Trees can be computationally expensive, particularly for large datasets [18].

iv. K-Nearest Neighbors (KNN) Classifier
The K-Nearest Neighbors (KNN) classifier is a supervised machine learning algorithm that classifies new data points based on the majority vote of their k-nearest neighbors. KNN is easy to implement and understand. It can manage non-linear relationships between features and is robust to noisy data. However, KNN can be computationally expensive for large datasets. Finding the optimal value of k can be challenging, and KNN is sensitive to feature scaling [19].

v. Neural Network Architecture
A neural network architecture is a type of machine learning model inspired by the structure and function of the human brain. It consists of interconnected nodes (neurons) arranged in layers that transform and process inputs to produce outputs. Neural networks can approximate any continuous function and handle noisy or missing data. They can also be parallelized for efficient computation. However, they require significant computational resources and may overfit training data if not properly regularized. Additionally, complex neural network models can be difficult to interpret [20].

vi. Xgboost Classifier
XGBoost (Extreme Gradient Boosting) is an open-source, supervised machine learning algorithm that uses gradient boosting to classify data. It is widely employed for classification and regression tasks due to its high performance, scalability, and interpretability. XGBoost is known for its accuracy and speed. It provides feature importance scores, facilitating result interpretation. Additionally, XGBoost can handle missing values without imputation and manage large datasets efficiently. However, it requires careful hyperparameter tuning to prevent overfitting and demands significant computational resources for large datasets [21].

vii. Random Forest (RF) Classifier
A Random Forest (RF) classifier is an ensemble learning algorithm that uses multiple decision trees for data classification. It is popular and widely used in machine learning due to its simplicity, interpretability, and effectiveness. RF often achieves high accuracy because of its ensemble approach. It can handle noisy data and outliers effectively. RF provides feature importance scores, making results easier to interpret. Additionally, RF can manage high-dimensional data with many features. However, RF can be computationally expensive for large datasets and may overfit if the number of decision trees is too high [22].

viii. Extra Trees Classifier (ETC).
The Extra Trees Classifier (ETC) is an ensemble learning method that uses multiple decision trees to classify data. It is similar to the Random Forest Classifier but with key differences. ETC trains faster than Random Forest because it does not use bootstrap sampling. It can handle noisy data and outliers effectively. ETC provides feature importance scores, which facilitate result interpretation. It is capable of managing high-dimensional data with many features. However, ETC may have lower accuracy than Random Forest due to the lack of bootstrap sampling. Additionally, ETC can overfit if the number of decision trees becomes too large [23].

To facilitate a clearer understanding, Table 2 presents a comparative summary of the main properties, advantages, and limitations of the discussed machine learning classifiers.

Table 2.
ML Classifiers Comparison.

Classifier	Advantages	Disadvantages	Common Applications
NB	- Fast and Simple - Works with less data - handle high dimensions	- "Naive" Assumption of feature independence - Often less accurate than complex models	- Text Classification - Sentiment Analysis - Real-time Prediction
SVM	- Effective in high dimensions - Versatile with different kernels	- Slow on large datasets - Not great with noisy or overlapping data	- Image Classification - Bioinformatics - Handwriting Recognition
Decision Tree	- Easy to interpret - Handles both numerical and categorical data - Non-parametric	- Prone to overfitting - Unstable; small data changes can alter the tree	- Credit Scoring - Customer Segmentation
KNN	- Simple and intuitive - No training phase - Adapts easily to new data	- Slow prediction time on large datasets - Performance degrades with many features	- Recommender Systems - Image Recognition - Anomaly Detection
Neural Network	- Learns complex, non-linear patterns - Can automatically learn features	- "Black box" nature makes it hard to interpret - Requires large amounts of data - Computationally expensive to train	- Image and Speech Recognition - Natural Language Processing (NLP) - Autonomous Driving
XGBoost	- high performance - Built-in regularization to prevent overfitting - Natively handles missing values	- Complex and more difficult to tune - Can be sensitive to hyperparameters	- Sales Forecasting - Fraud Detection
RF	- High accuracy and robust to overfitting - Handles large datasets efficiently - Provides feature importance scores	- Less interpretable than a single decision tree - Can be slow to make predictions if it has many trees	- Banking (credit risk) - Stock Market Prediction
ETC	- Very fast to train - Added randomness can reduce variance	- Increased randomness might slightly increase bias	- Feature Selection

4. Methodology

The proposed methodology for human activity recognition is organized into five main stages, as shown in Figure 3. The process begins with data acquisition, where motion signals are collected using smartphone sensors such as the accelerometer and gyroscope. The collected data then undergoes preprocessing, which includes cleaning and label encoding to ensure consistency and noise removal. During the model training phase, the processed dataset is split into training and testing subsets, and the Extra Trees Classifier is used to learn activity patterns. The trained model is subsequently evaluated in the model evaluation stage using key performance metrics such as accuracy, precision, recall, and F1-score. Finally, the activity recognition stage interprets the model outputs to identify various human activities. This structured workflow provides a systematic and reproducible approach for accurate activity classification.

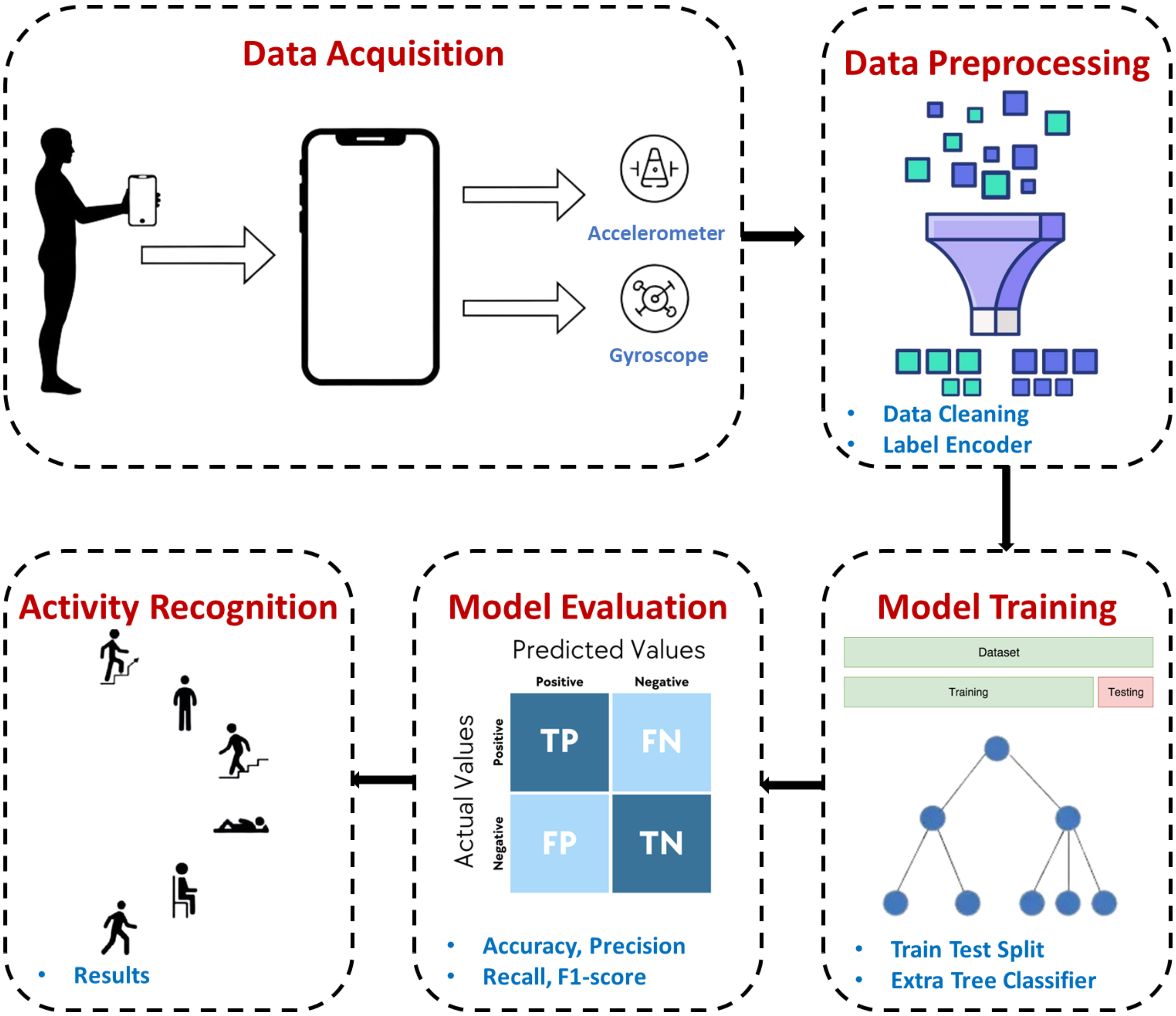

Figure 1.
Methodology.

This section outlines the methodological framework employed in this study, encompassing data acquisition, preprocessing, and model evaluation. It begins with a detailed description of the dataset used, including its structure, sources, and relevant attributes. The subsequent subsection discusses the preprocessing techniques applied to prepare the data for model training, such as cleaning, normalization, and encoding. Finally, the evaluation measures used to assess the performance and reliability of the proposed models are presented. Together, these components establish a systematic approach that ensures the robustness and validity of the experimental results.

4.1. Data Set Description

The experiments were based on the HARSense dataset, which contains subject-wise daily living activity data collected from smartphone gyroscope and accelerometer sensors. The smartphone was fixed on users' front and waist pockets. Running was performed on a football playground, while all other activities took place in a laboratory. The dataset comprises 17 columns and 94,198 rows. The activities include walking, standing, upstairs, downstairs, running, sleeping, and sitting. The column descriptors are: RV (Rotational Vector in x, y, and z axes), RR (Rotational Rate in x, y, and z axes), Gravity (Gravity in x, y, and z axes), Acc (Linear Acceleration in x, y, and z axes), and AG (Acceleration due to Gravity in x, y, and z axes) [24]. Table 3 provides detailed descriptions of the dataset features.

Table 3.
HARSense Data Features Description.

Column	Description	Sensor/Origin
AG-X, AG-Y, AG-Z	**Angular Gyroscope**. These values represent the rate of rotation (angular velocity) of the device around its X, Y, and Z axes.	Gyroscope
Acc-X, Acc-Y, Acc-Z	**Total Acceleration**. This is the raw acceleration measured along the X, Y, and Z axes. It includes both the force of gravity and the linear acceleration caused by the user's motion.	Accelerometer
Gravity-X, Gravity-Y, Gravity-Z	**Gravity Vector**. This is the isolated gravity component of the total acceleration. It indicates the direction of "down" relative to the device's coordinate system.	Sensor Fusion
RR-X, RR-Y, RR-Z	**Rotation Rate**. It measures the speed of rotation around the X, Y, and Z axes.	Gyroscope
RV-X, RV-Y, RV-Z	**Rotation Vector**. This is a composite value representing the device's orientation in space. It is derived by fusing data from the accelerometer and gyroscope. The three values are components of a vector, with the direction indicating the axis of rotation and the magnitude indicating the angle of rotation.	Sensor Fusion
Cos	**Cosine of an Angle**. the cosine of the angle between the device's main axis and the vertical (gravity) vector.	Calculated Feature
Activity	**Activity Label**. This is the target variable you are trying to predict. It is a categorical label describing the physical activity being performed at that moment (e.g., 'walking', 'sitting', 'running').	Ground Truth

Figure 2 illustrates the distribution of data samples across seven activity classes in the dataset. The most common activities are 'Walking' (25,371 samples) and 'Sitting' (19,784 samples), forming the majority. Conversely, activities like 'downstairs' (8,487 samples) and especially 'Sleeping' (2,377 samples) are significantly underrepresented.

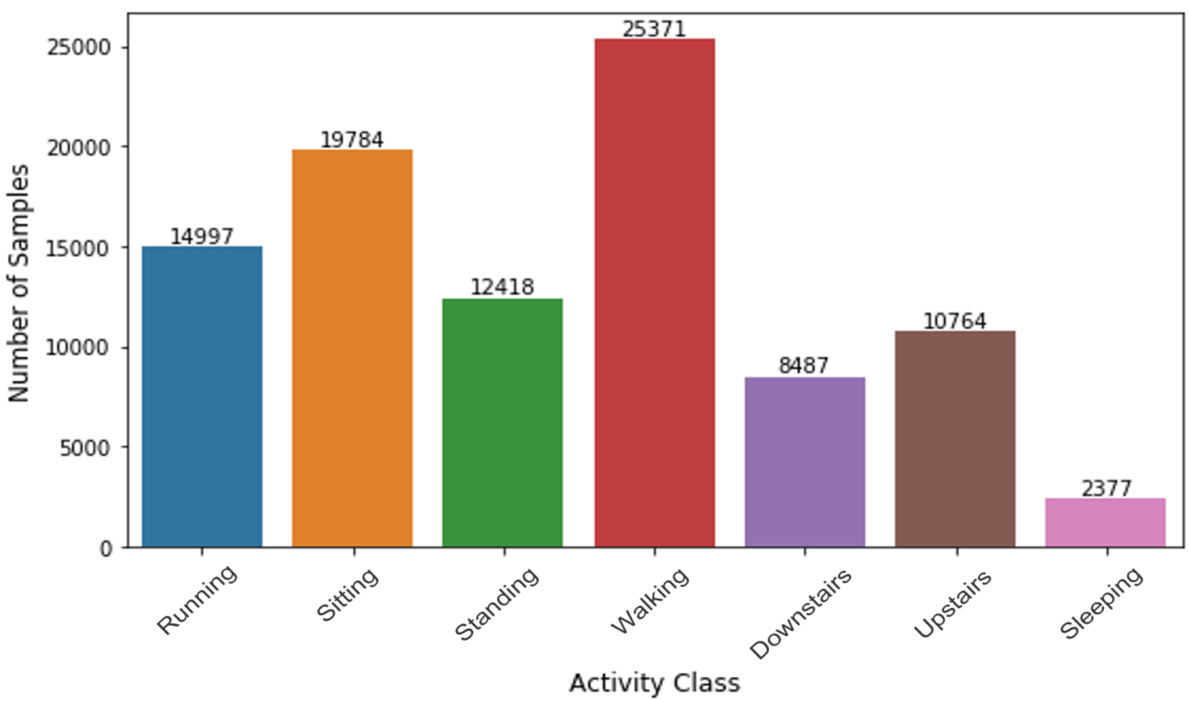

Figure 2.
Activity Target Distribution.

4.2. Data Preprocessing

To ensure the integrity and suitability of the HAR Sense dataset for machine learning applications, comprehensive data preprocessing steps were implemented to enhance data quality and model performance.

During the data preprocessing stage, the dataset was examined for missing and incomplete values. The "Acc-Y" column contained several missing values. To maintain data consistency and prevent errors during model training, these missing values were imputed with zeros using the command. This approach assumes that the absence of sensor readings corresponds to a neutral state (no acceleration) along the Y-axis, which is reasonable for inertial sensor data.

An exploratory data analysis was conducted to understand the characteristics of the sensor features, with the resulting distributions visualized in Figure 3. The gyroscopic measurements (AG- and RR-prefixes) exhibit sharp, leptokurtic distributions centered at zero, indicating that the device was predominantly static, with high-velocity movements appearing as outliers. In contrast, the total acceleration (Acc-) features show a wider variance, characteristic of dynamic human motion. The gravity vector components reveal a primary device orientation, with Gravity-Y heavily skewed toward 9.8 m/s^2. The multimodal distributions of the Rotation Vector (RV-) components and the cos feature confirm that a diverse range of device orientations and user postures were captured. This variety is essential for training a robust and generalizable activity recognition model.

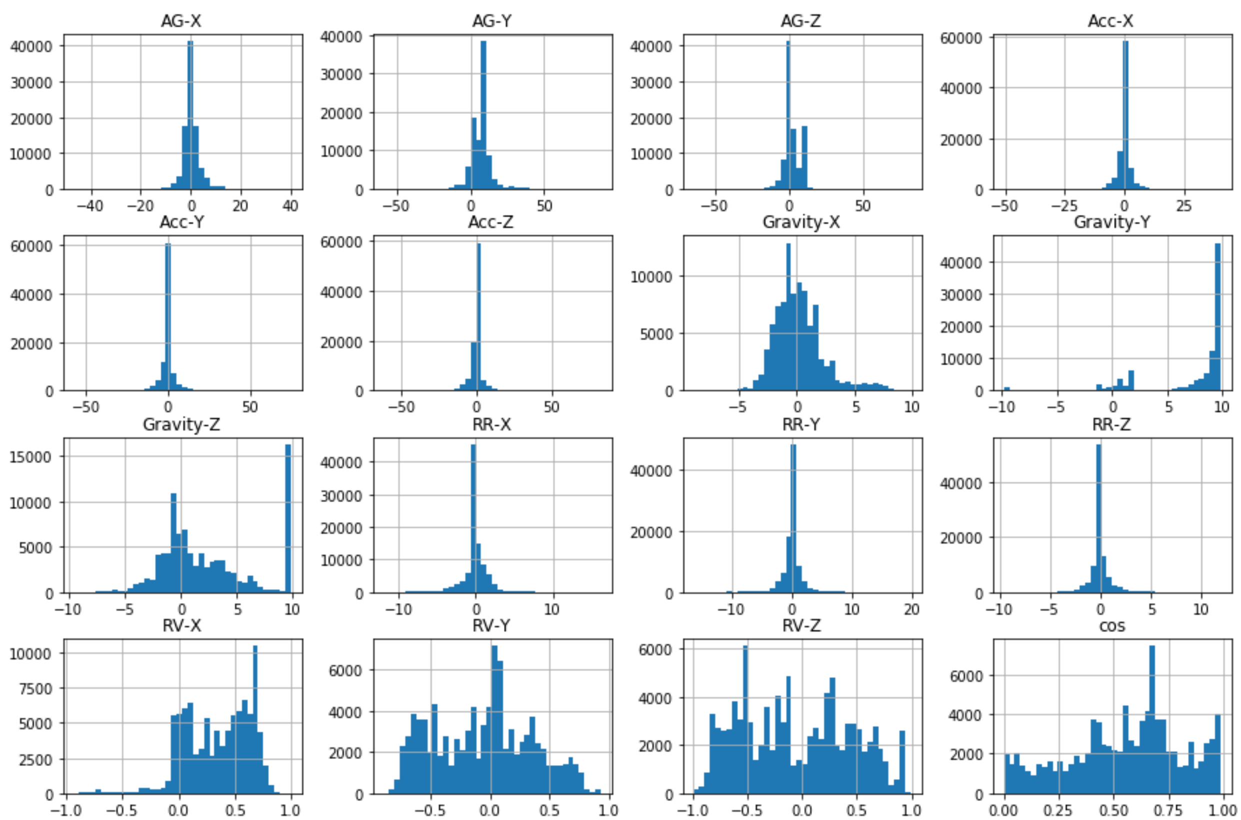

Figure 3.
Features Distribution.

First, non-informative columns were removed to reduce dimensionality and noise, aligning with data cleaning best practices. Missing values in the dataset were imputed with zeros to prevent adverse effects on model training. Outlier analysis was conducted using statistical methods to identify anomalous data points that could skew model learning. These steps aimed to improve data quality and model performance by systematically addressing irrelevant features, missing data, and outliers.

Next, the encoding of categorical variables was performed. Activity labels, being categorical, were converted into a numerical format using label encoding techniques. This step is essential for machine learning algorithms to process categorical data effectively. Each activity was assigned a numeric value (Table 4).

Table 4.
Numeric value association for each activity.

Activity	Numerical assignment
Running	0
Sitting	1
Sleeping	2
Standing	3
Walking	4
Downstairs	5
Upstairs	6

Following the cleaning and transformation steps, the dataset was structured to facilitate machine learning modeling. This involved organizing the data into a format suitable for algorithmic processing, ensuring that all features were appropriately scaled and encoded.

Finally, to evaluate model performance, the dataset was divided into training and testing subsets, with 80% allocated for training and 20% for testing.

4.3. Evaluation Measures

The performance of the proposed models is evaluated using multiple measures, which are accuracy, precision, recall, and F1-score. These measures are briefly explained in the following paragraphs

A) Accuracy is the ratio of correct predictions to total predictions, calculated as Accuracy = (TP + TN) / (TP + FP + FN + TN). It is useful when classes are balanced, but it can be misleading with imbalanced classes [25].
B) Precision is the number of correct classes returned by the classification model, calculated as Precision = TP/(TP+FP). It is useful when the cost of false positives is high [25].
C) Recall is the ability of a model to find all relevant cases within a dataset, and it is calculated as: Recall = TP/(TP+FN). Recall is useful when the cost of false negatives is high [25].
D) F1-Score is the harmonic mean of precision and recall and can be determined as: F1 = 2 *Precision * Recall/(Precision + Recall) or F1 = 2*TP/(2*TP + FP + FN). The F1-score provides a balanced measure of both precision and recall [25].

5. Experiment Results

All experiments were conducted using Google Colaboratory (Colab), a cloud-based platform that provides a flexible and efficient environment for developing and testing machine learning models. The Colab environment was configured to run on Ubuntu 22.04.4 LTS (64-bit) with Python 3.10. The computational setup included an Intel(R) Xeon(R) CPU @ 2.20 GHz, 12 GB of RAM, and an optional NVIDIA Tesla T4 GPU with 16 GB VRAM, which was utilized to accelerate training processes where applicable.

Multiple experiments have been conducted to examine the performance of individual classifiers, such as Neural Network architecture, Naïve Bayes, decision trees, SVM, and KNN classifiers. The results of these classifiers were compared with those obtained from ensemble learning techniques, including random forest, extra trees, and XGB.

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 10, No. 1: 1119-1137, 2026
DOI: 10.55214/2576-8484.v10i1.11864

In these experiments, the dataset was divided into 80% for training and 20% for testing. The results of the classical individual classifiers are listed in Table 5, and Figure 4 shows their accuracy. The results of the ensemble classifiers are in Table 6, with Figure 5 illustrating their accuracy. Precision, recall, and F1-score were calculated for each activity, followed by overall accuracy. Both macro-average and weighted-average metrics were computed; the macro-average treats all activities equally, while the weighted-average assigns weights based on the ratio of samples per activity.

In addition to the quantitative performance metrics, the confusion matrices for both the individual and ensemble classifiers are presented in Figures 4 and 5, respectively. These matrices offer a detailed visualization of each model's classification performance across different activity categories, highlighting correctly and incorrectly predicted instances.

Table 5.
Classical individual classifiers' results for all activities.

	Activity no.	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**macro avg.**	**weighted avg.**
knn	Precision	0.95	0.99	0.99	0.98	0.9	0.88	0.86	0.94	0.93
	Recall	0.9	1	1	0.99	0.96	0.78	0.84	0.93	0.93
	F1-score	0.92	1	0.99	0.99	0.93	0.83	0.85	0.93	0.93
	Accuracy	0.9341								
svm	Precision	0.92	0.96	0.99	0.85	0.71	0.83	0.82	0.87	0.85
	Recall	0.88	1	1	0.96	0.93	0.43	0.39	0.8	0.84
	F1-score	0.9	0.98	0.99	0.9	0.8	0.57	0.53	0.81	0.82
	Accuracy	0.8358								
Decision tree	Precision	0.9	0.99	1	0.99	0.91	0.8	0.78	0.91	0.91
	Recall	0.88	1	1	0.99	0.91	0.8	0.8	0.91	0.91
	F1-score	0.89	1	1	0.99	0.91	0.8	0.79	0.91	0.91
	Accuracy	0.9132								
Naive Bayes	Precision	0.83	0.95	1	0.77	0.57	0.49	0.46	0.72	0.71
	Recall	0.66	0.86	0.98	0.9	0.89	0.15	0.22	0.67	0.71
	F1-score	0.74	0.91	0.99	0.83	0.69	0.23	0.3	0.67	0.68
	Accuracy	0.7074								
Neural network	Precision	0.94	1	1	0.99	0.93	0.85	0.87	0.94	0.94
	Recall	0.93	1	1	0.99	0.95	0.83	0.83	0.93	0.94
	F1-score	0.94	1	1	0.99	0.94	0.84	0.85	0.94	0.94
	Accuracy	0.9406								

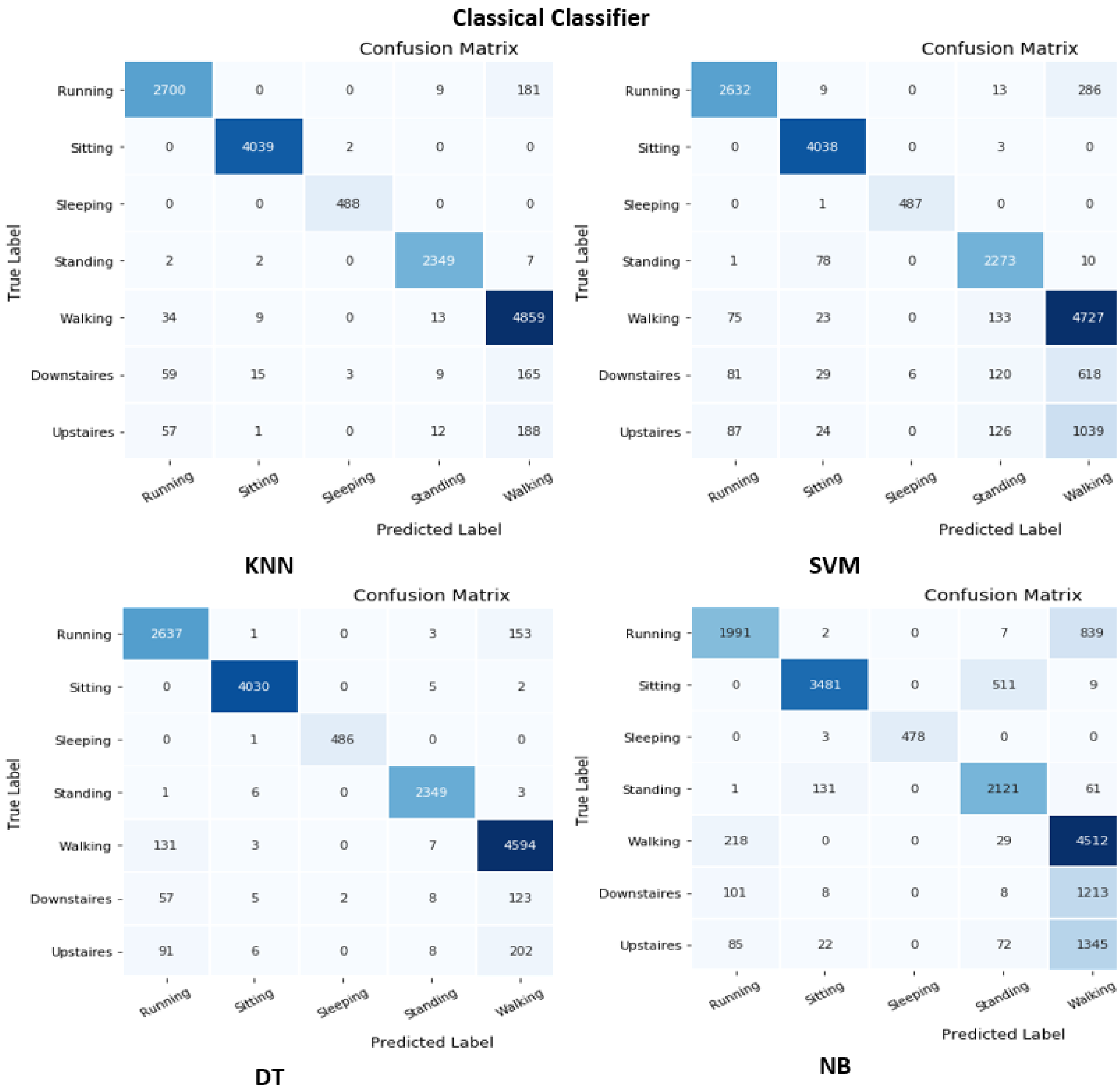

Figure 4.
Confusion matrices of classical machine learning algorithms.

Table 6.
Ensemble classifiers' results for all activities.

	Activity No.	0	1	2	3	4	5	6	macro avg.	weighted avg.
XG Boost	precision	0.96	1	1	1	0.94	0.92	0.89	0.96	0.96
	recall	0.96	1	1	1	0.96	0.87	0.88	0.95	0.95
	f1-score	0.96	1	1	1	0.95	0.9	0.89	0.96	0.96
	accuracy	0.958								
Random Forest	precision	0.94	1	1	1	0.95	0.97	0.91	0.96	0.97
	recall	0.96	1	1	1	0.97	0.85	0.91	0.96	0.96
	f1-score	0.95	1	1	1	0.96	0.91	0.91	0.96	0.96
	accuracy	0.9625								
Extra_trees	precision	0.96	1	1	1	0.96	0.97	0.93	0.97	0.97
	recall	0.97	1	1	1	0.98	0.89	0.92	0.96	0.97
	f1-score	0.97	1	1	1	0.97	0.93	0.93	0.97	0.97
	accuracy	0.9712								

It can be observed from the results of the experiments conducted using individual classifiers that the KNN classifier achieved a high accuracy of 93.41%. The F1-scores for all activities exceed 0.90, although activity 5 has an F1-score of 0.83, possibly due to overlapping features with other activities. Among all classifiers, Naive Bayes had the lowest accuracy at 70.74%, indicating difficulties in modeling complex activity patterns. Conversely, the neural network achieved the highest accuracy of 94.06%, owing to its capacity to model complex feature relationships. However, neural networks require proper tuning to prevent overfitting.

The results of the conducted experiments show that ensemble learning methods significantly outperform individual classifiers. Two classifiers, KNN and Neural Networks, achieved high accuracies of 93.41% and 94.06%, respectively. Conversely, Naive Bayes and SVM had the lowest accuracies of 70.74% and 83.58%, respectively, with difficulties in identifying Activities 5 and 6. The decision tree classifier achieved a moderate accuracy of 91.32%.

Ensemble learning classifiers, including XGBoost, Random Forest, and Extra Trees, achieved superior accuracies of 95.81%, 96.25%, and 97.12%, respectively. These models outperformed individual classifiers. Extra Trees yielded the highest accuracy, macro average F1-score, and weighted average F1-score at 97.12%, 97%, and 97%, respectively. Most classifiers accurately recognized Activities 1, 2, and 3, but faced difficulties with Activities 5 and 6, particularly Naive Bayes. These findings confirm that ensemble models outperform individual classifiers in HAR recognition by combining multiple models to enhance prediction robustness.

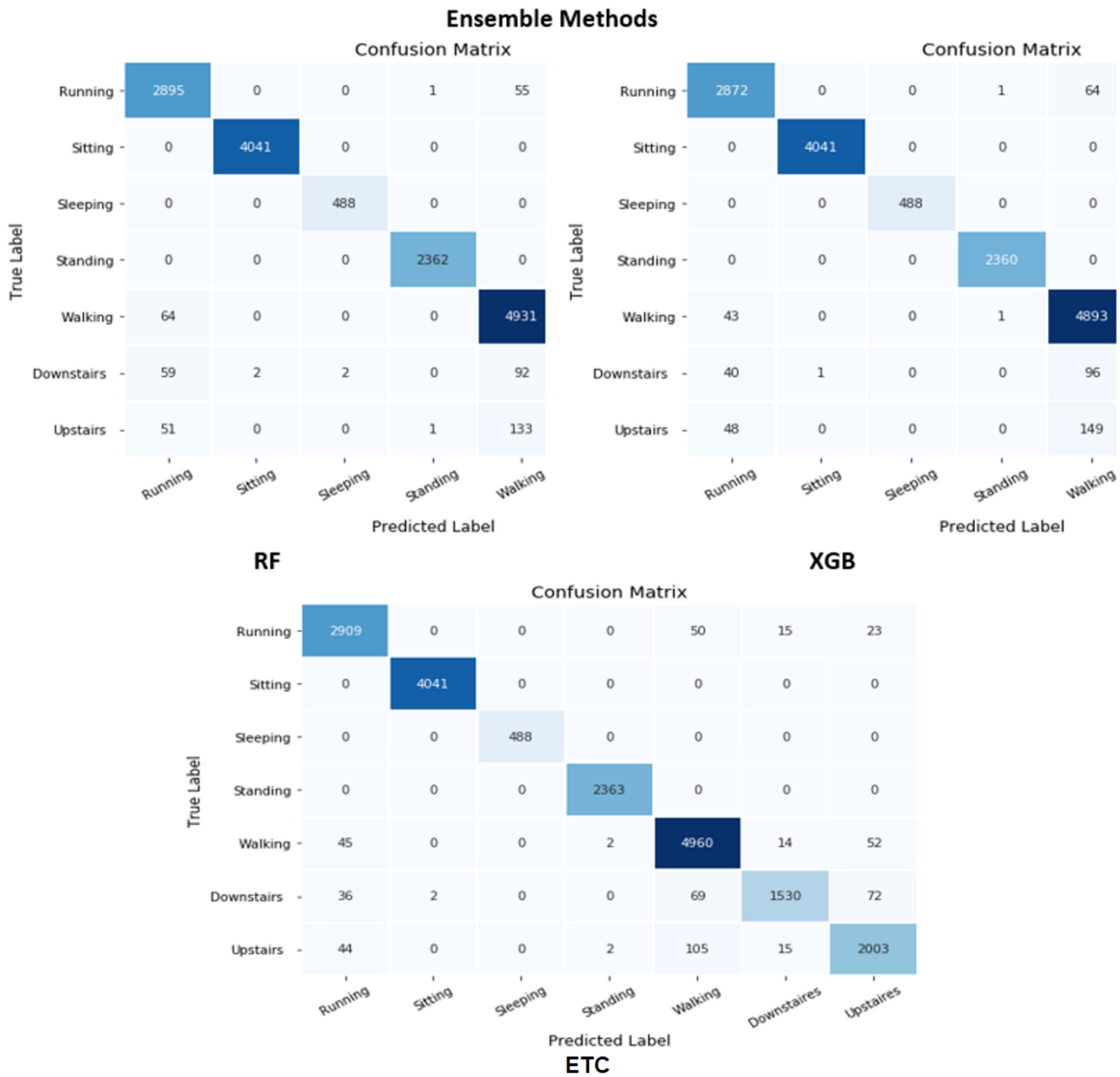

Figure 5.
Confusion matrices of ensemble machine learning algorithms.

Table 7 summarizes the results from all classifiers regarding our experiment based on the HARSense dataset, presenting accuracy, macro average, and weighted average for precision, recall, and F1-score. Figures 6 and 8 illustrate the accuracy, macro, and weighted averages for precision, recall, and F1-score, respectively, from all applied classifiers.

Our accuracy results are compared against those from Nematallah and Rajan [15], who used decision tree (DT) and support vector machines (SVM), achieving 74.62% and 76.53%, respectively, with the coif14-based wavelet packet transform for the HARSense dataset. Our findings indicate that the most effective machine learning technique for human activity analysis is the extra trees classifier, with an accuracy of 97.12%.

Table 7.
Summary results for the proposed methods.

Classifier	Accuracy	Precision		Recall		F1-Score	
		Macro Avg	Weighted Avg	Macro Avg	Weighted Avg	Macro Avg	Weighted Avg
Naive_bayes	0.7074	0.72	0.71	0.67	0.71	0.67	0.68
svm	0.8358	0.87	0.87	0.80	0.84	0.81	0.82
Decision_tree	0.9132	0.91	0.91	0.91	0.91	0.91	0.91
knn	0.9341	0.94	0.93	0.93	0.93	0.93	0.93
Neural network architecture keras	0.9406	0.94	0.94	0.93	0.94	0.94	0.94
XGBoost	0.9580	0.96	0.96	0.95	0.96	0.96	0.96
Random_forest	0.9625	0.97	0.96	0.96	0.96	0.96	0.96
Extra_trees	0.9712	0.97	0.97	0.97	0.97	0.97	0.97

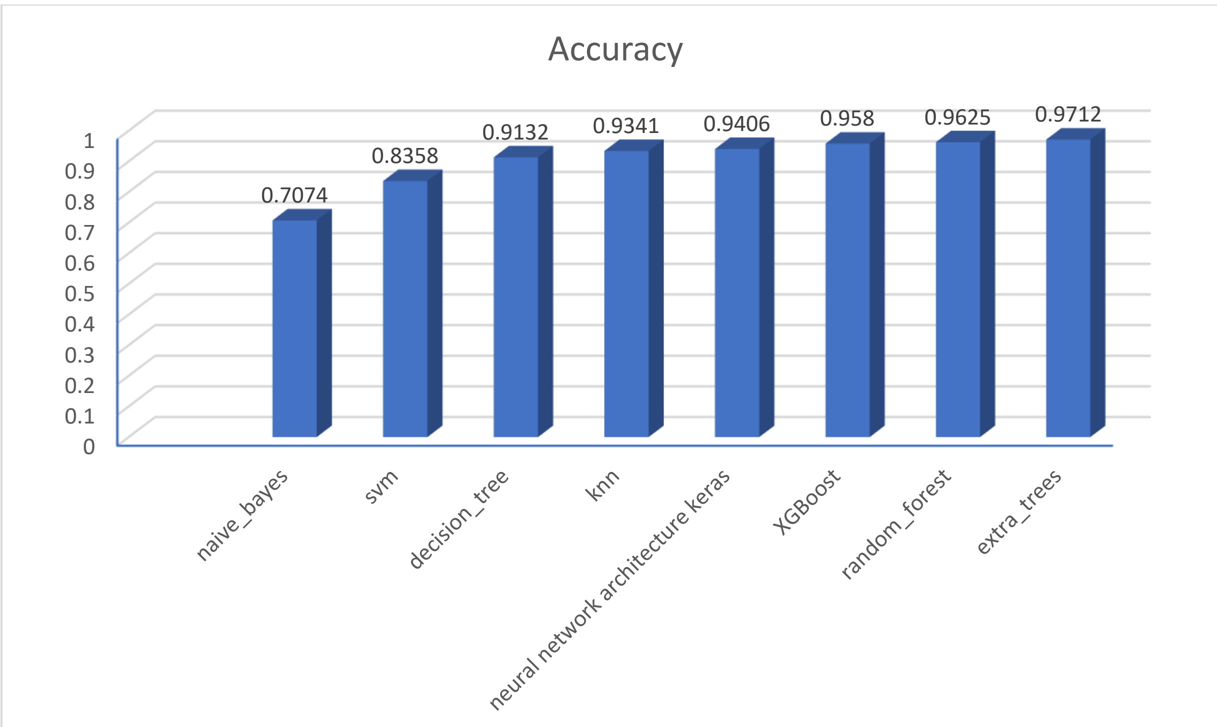

Figure 6.
Accuracy results from all applied classifiers.

Edelweiss Applied Science and Technology
ISSN: 2576-8484
Vol. 10, No. 1: 1119-1137, 2026
DOI: 10.55214/2576-8484.v10i1.11864

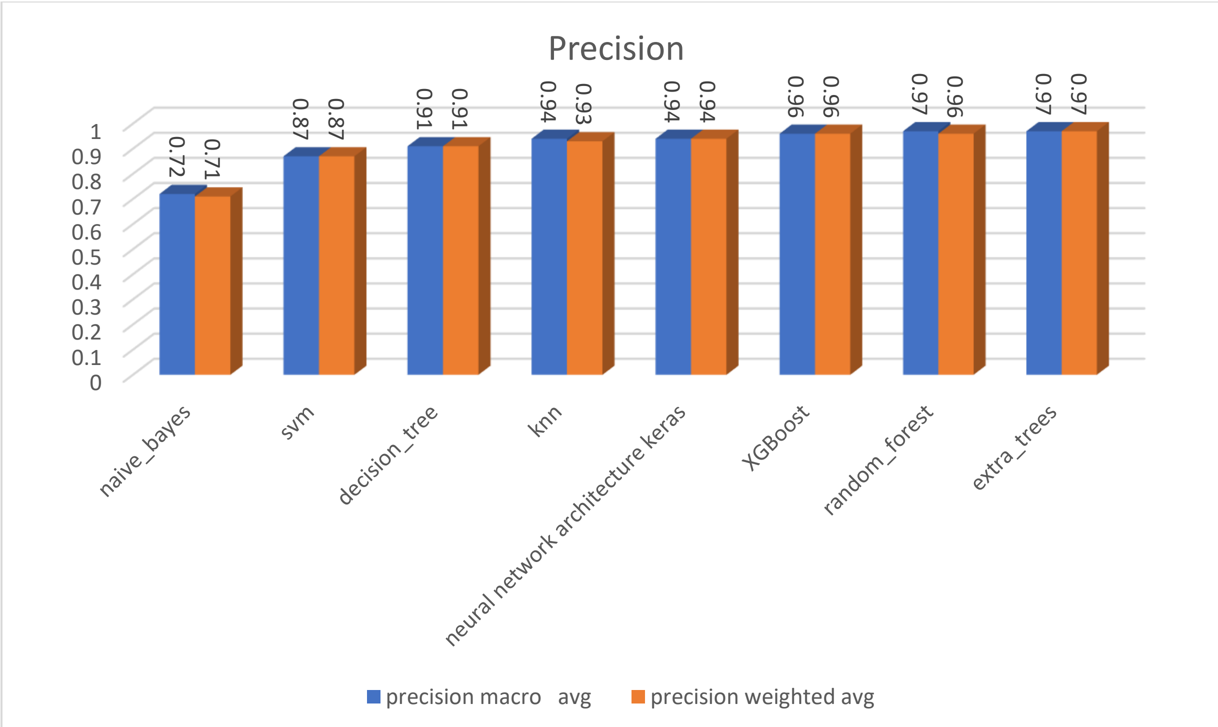

Figure 1.
Precision macro and weighted average results from all applied classifiers.

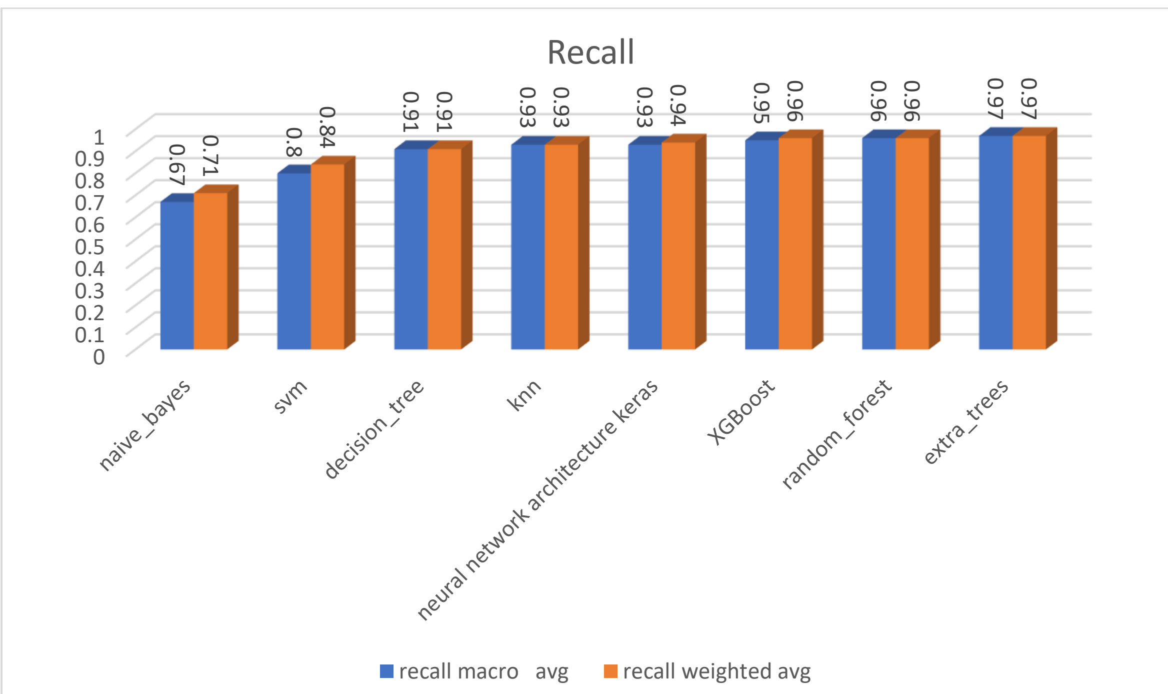

Figure 8.
Recall macro and weighted average results from all applied classifiers

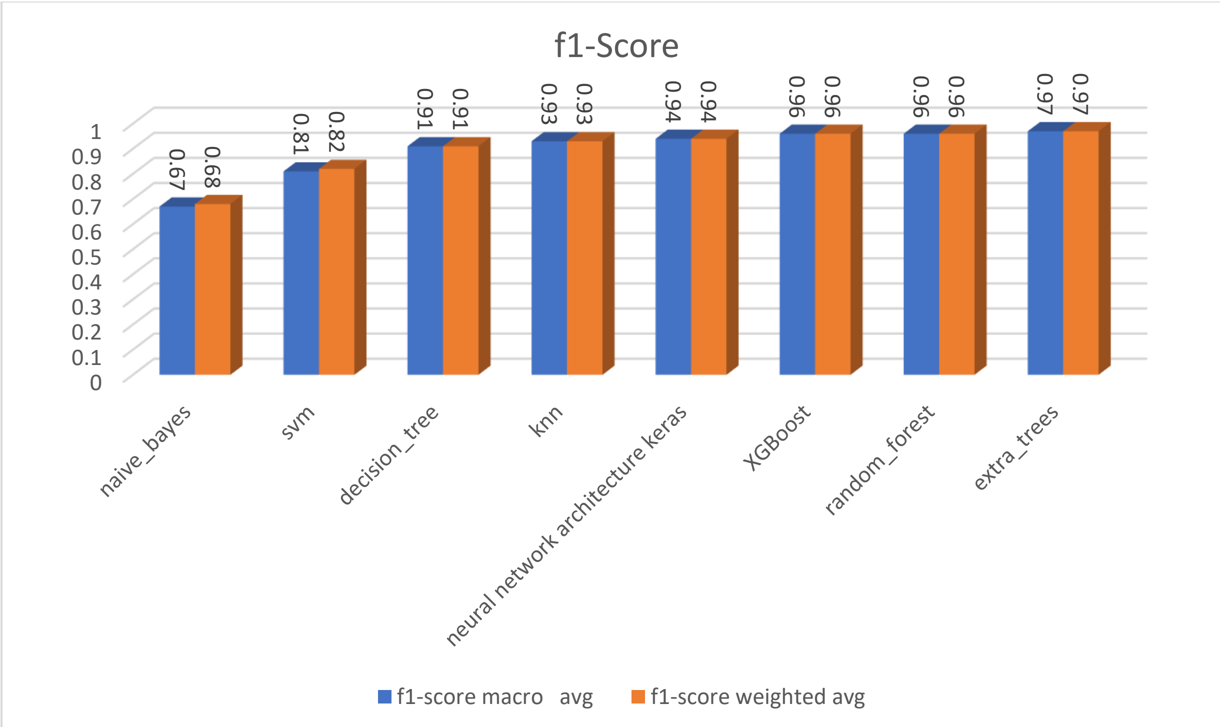

Figure 9.
f1-Score macro and weighted average results from all applied classifiers.

6. Conclusion

The HARSense dataset provides an up-to-date resource for researchers in this domain. We conducted experiments using various machine learning techniques, including the Naive Bayes classifier, SVM classifier, decision tree classifier, KNN classifier, neural network architecture, XGB classifier, random forest classifier, and extra trees classifier. We achieved promising results with the extra trees classifier. In future work, we plan to explore other deep learning techniques on the HARSense dataset and compare their performance. The most effective machine learning method evaluated for human activity analysis was the Extra Trees classifier, with an accuracy of 97.12%.

Data Availability:

Data that support the findings of this study are available at the following link: https://www.kaggle.com/datasets/nurulaminchoudhury/harsense-datatset.

Transparency:

The authors confirm that the manuscript is an honest, accurate, and transparent account of the study; that no vital features of the study have been omitted; and that any discrepancies from the study as planned have been explained. This study followed all ethical practices during writing.

References

[1] N. A. Choudhury, S. Moulik, and D. S. Roy, "Harsense: statistical human activity recognition dataset," *IEEE Dataport*, 2021.

[2] M. M. Hassan, M. Z. Uddin, A. Mohamed, and A. Almogren, "A robust human activity recognition system using smartphone sensors and deep learning," *Future Generation Computer Systems*, vol. 81, pp. 307-313, 2018. https://doi.org/10.1016/j.future.2017.11.029

[3] S. Singh, N. A. Choudhury, and B. Soni, "Gait recognition using activities of daily livings and ensemble learning models," in *International Conference on Advances in IoT and Security with AI (pp. 195-206). Singapore: Springer Nature Singapore*, 2023.

[4] A. A. Wazwaz, K. M. Amin, N. A. Semari, and T. F. Ghanem, "Enhancing human activity recognition using features reduction in IoT edge and Azure cloud," *Decision Analytics Journal*, vol. 8, p. 100282, 2023. https://doi.org/10.1016/j.dajour.2023.100282

[5] A. O. Ige and M. H. M. Noor, "A survey on unsupervised learning for wearable sensor-based activity recognition," *Applied Soft Computing*, vol. 127, p. 109363, 2022. https://doi.org/10.1016/j.asoc.2022.109363

[6] M. Z. Uddin and A. Soylu, "Human activity recognition using wearable sensors, discriminant analysis, and long short-term memory-based neural structured learning," *Scientific Reports*, vol. 11, p. 16455, 2021. https://doi.org/10.1038/s41598-021-95947-y

[7] U. Verma, P. Tyagi, and M. Kaur, "Artificial intelligence in human activity recognition: A review," *International Journal of Sensor Networks*, vol. 41, no. 1, pp. 1-22, 2023. https://doi.org/10.1504/IJSNET.2023.128503

[8] N. Gupta, S. K. Gupta, R. K. Pathak, V. Jain, P. Rashidi, and J. S. Suri, "Human activity recognition in artificial intelligence framework: A narrative review," *Artificial Intelligence Review*, vol. 55, pp. 4755-4808, 2022. https://doi.org/10.1007/s10462-021-10116-x

[9] Z. H. Chin, H. Ng, T. T. V. Yap, H. L. Tong, C. C. Ho, and V. T. Goh, *Daily activities classification on human motion primitives detection dataset. Computational Science and Technology: 5th ICCST 2018, Kota Kinabalu, Malaysia, 29-30 August 2018.* Singapore: Springer Singapore, 2018.

[10] B. Padmaja, V. Prasad, and K. Sunitha, "A novel random split point procedure using extremely randomized (Extra) trees ensemble method for human activity recognition," *EAI Endorsed Transactions on Pervasive Health & Technology*, vol. 6, no. 22, pp. 1-10, 2020. https://doi.org/10.4108/eai.28-5-2020.164824

[11] T. F. N. Bukht, H. Rahman, and A. Jalal, "A decision tree-based framework for human action recognition by using feature fusion," *Computer*, vol. 3, p. 5, 2023.

[12] S. Khan, S. H. Noorani, A. Arsalan, A. Mahmood, U. Rauf, and Z. Ali, "Classification of human physical activities and postures during everyday life," in *2023 18th International Conference on Emerging Technologies (ICET) (pp. 98-103). IEEE*, 2023.

[13] S. Khan *et al.*, "A framework for daily living activity recognition using fusion of smartphone inertial sensors data," in *2023 4th International Conference on Computing, Mathematics and Engineering Technologies (iCoMET) (pp. 1-6). IEEE*, 2023.

[14] H. Zhu, J. Guo, Z. Shao, Y. Zhang, Z. Ding, and H. Gong, "Extra-trees for human activity recognition using wearable sensors," in *2022 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS) (pp. 801-804). IEEE*, 2022.

[15] H. Nematallah and S. Rajan, "Quantitative analysis of mother wavelet function selection for wearable sensors-based human activity recognition," *Sensors*, vol. 24, no. 7, p. 2119, 2024. https://doi.org/10.3390/s24072119

[16] A. McCallum, "Graphical models, lecture2: Bayesian network represention," *PDF). Retrieved*, vol. 22, 2019.

[17] A. Singh and S. Sharma, "Analysis on data mining models for internet of things," presented at the In 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), 2017, pp. 94–100, 2017.

[18] M. Nalluri, M. Pentela, and N. R. Eluri, "A scalable tree boosting system: XG boost," *International Journal of Research Studies in Science, Engineering and Technology*, vol. 7, no. 12, pp. 36-51, 2020.

[19] M. Suyal and P. Goyal, "A review on analysis of k-nearest neighbor classification machine learning algorithms based on supervised learning," *International Journal of Engineering Trends and Technology*, vol. 70, no. 7, pp. 43-48, 2022.

[20] S. Walczak, *Artificial neural networks. In Encyclopedia of information science and technology.* Hershey, PA: IGI Global, 2018.

[21] Z. A. Ali, Z. H. Abduljabbar, H. A. Tahir, A. B. Sallow, and S. M. Almufti, "eXtreme gradient boosting algorithm with machine learning: A review," *Academic Journal of Nawroz University*, vol. 12, no. 2, pp. 320-334, 2023. https://doi.org/10.25007/ajnu.v12n2a1612

[22] A. N. S. Kinasih, A. N. Handayani, J. T. Ardiansah, and N. S. Damanhuri, "Comparative analysis of decision tree and random forest classifiers for structured data classification in machine learning," *Science in Information Technology Letters*, vol. 5, no. 2, pp. 13-24, 2024.

[23] A. Sanmorino, L. Marnisah, and H. Sunardi, "Feature selection using extra trees classifier for research productivity framework in Indonesia," in *Proceeding of the 3rd International Conference on Electronics, Biomedical Engineering, and Health Informatics: ICEBEHI 2022, 5–6 October, Surabaya, Indonesia, 2023, pp. 13–21*, 2023.

[24] N. A. Choudhury, S. Moulik, and D. S. Roy, "Physique-based human activity recognition using ensemble learning and smartphone sensors," *IEEE Sensors Journal*, vol. 21, no. 15, pp. 16852-16860, 2021. https://doi.org/10.1109/JSEN.2021.3077563

[25] M. Muntean and F.-D. Militaru, "Metrics for evaluating classification algorithms," in *Education, Research and Business Technologies: Proceedings of 21st International Conference on Informatics in Economy (IE 2022)*, 2023.

DOI: 10.55214/2576-8484.v10i1.11864