Comparative Study of Deep Learning Models for Human Activity Recognition

Abstract

Human Activity Recognition (HAR) using wearable sensor data is a cornerstone of mobile health and context-aware computing. While deep learning has significantly advanced HAR accuracy the computational demands of complex architectures often conflict with the limited resources of edge devices like smartphones and wearables . This creates a critical trade-off between predictive performance and practical deployability. This paper presents a systematic comparative analysis of five distinct deep learning architectures: a baseline Multi-Layer Perceptron (MLP), a 1D Convolutional Neural Network (1D-CNN), a Long Short-Term Memory (LSTM) network, a hybrid CNN-LSTM model , and a Transformer-based model . We evaluate these models on the public UCI-HAR dataset, focusing not only on classification accuracy and F1-score but also on crucial efficiency metrics: model size (parameters) and inference latency. Our findings reveal that while the Transformer achieves the highest F1-score (0.931), its substantial computational cost makes it less suitable for real-time edge applications. The hybrid CNN-LSTM architecture emerges as the most balanced solution, delivering competitive accuracy (0.925 F1-score) with significantly lower latency and a more compact model size. This study provides a clear, data-driven framework for selecting appropriate HAR models based on specific deployment constraints.

Keywords: Human Activity Recognition(HAR), Convolution Neural Network(CNN), Recurrent Neural Network(RNN), Support Vector Machine(SVM), Multilayer Perceptron (MLP)

1. Introduction

The proliferation of sensor-rich mobile and wearable devices has catalyzed research in Human Activity Recognition (HAR) [1]. By interpreting data from accelerometers and gyroscopes, HAR systems can enable a host of applications, from remote patient monitoring and elderly care to fitness tracking and smart home automation [1].

Historically, HAR systems relied on handcrafted feature engineering coupled with traditional machine learning classifiers like Support Vector Machines (SVMs) [2]. This approach, while effective, is labor-intensive and requires significant domain expertise. The advent of deep learning has revolutionized the field by enabling end-to-end learning, where models automatically extract hierarchical features directly from raw sensor data [3]. Architectures like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) have become the de facto standard, consistently achieving state-of-the-art results.

However, pushing the accuracy frontier has often led to increasingly complex and computationally expensive models [3]. This poses a significant challenge for real-world deployment, where HAR inference must occur in real-time on resource-constrained edge devices with limited battery life and processing power. A model that achieves 99% accuracy but drains a smartphone battery in an hour is impractical. This highlights a critical research gap: a holistic comparison that evaluates deep learning architectures not just on their predictive power but also on their operational efficiency [3].

1.1. Research Contribution:

1. A systematic implementation and evaluation of five architectures (MLP, 1D-CNN, LSTM, CNN-LSTM, Transformer) for sensor-based HAR

2. A holistic analysis balancing performance metrics (Accuracy, F1-Score) with efficiency metrics crucial for edge deployment (Model Parameters, Inference Latency)

3. A qualitative error analysis and a visual trade-off analysis

4. All evaluations are conducted on the well-established, public UCI-HAR benchmark dataset to ensure reproducibility and comparability

2. Related Work

Table 1 : Research Analysis

Sr. No.	Author	Findings	Limitations	Conclusion
1.	Ahmed, S. et al. [1]	AI models with wearable sensors provide promising solutions for HAR applications	Limited to specific sensor configurations and controlled environments	Wearable sensors integrated with AI models show significant potential for advancing HAR systems
2.	Anderson, T., et al. [2]	Holistic evaluation framework needed for HAR models considering both performance and computational efficiency	Focus primarily on model performance without extensive real-world deployment testing	Deep learning-based HAR requires balanced evaluation of accuracy and practical deployability constraints
3.	Chen, X., et al. [3]	Transformer-based models with attention mechanisms achieve high accuracy for HAR tasks	High computational requirements may limit edge device deployment	TCN-attention mechanisms provide superior performance but require consideration of resource constraints
4.	Garcia, L., et al.[4]	Deep learning enables end-to-end learning from raw sensor data, eliminating need for manual feature engineering	Limited evaluation on diverse real-world scenarios and noise conditions	Deep learning approaches significantly advance HAR by automating feature extraction processes
5.	Kaur, P., et al. [5]	Comprehensive overview of HAR field showing evolution from traditional to deep learning approaches	Primarily review-based without novel algorithmic contributions	HAR field has evolved significantly with deep learning becoming the dominant paradigm
6.	Kim, Y., et al. [6]	CNN-LSTM hybrid models enable effective feature extraction and temporal modeling for wearable sensor-based HAR	Limited to specific activity types and controlled experimental conditions	Hybrid architectures combining CNN and LSTM provide balanced approach for HAR applications

7.	Kumar, S., et al. [7]	Hybrid CNN-LSTM architecture provides effective approach for HAR applications, particularly in medical emergency scenarios	Focused on medical emergency contexts, may not generalize to general HAR applications	Bi-directional LSTM combined with CNN shows promise for critical healthcare applications
8.	Miller, K., et al. [8]	Deep learning models, particularly 1D-CNNs, are highly effective for wearable sensor-based HAR with good efficiency	Limited comparison with other deep learning architectures	1D-CNNs provide optimal balance between accuracy and computational efficiency for HAR
9.	Qin, Z., et al. [9]	Deep learning techniques show superior performance compared to traditional machine learning for smartphone and wearable sensor HAR	Review-based study without extensive experimental validation	Deep learning represents significant advancement over traditional approaches in HAR domain
10.	Ravi, D., et al. [10]	Resource efficiency is critical for HAR deployment on low-power edge devices, highlighting accuracy-efficiency trade-offs	Early work with limited deep learning architecture exploration	Established importance of considering computational constraints in HAR model development
11.	Rodriguez, M., et al. [11]	Deep neural networks can achieve device position-independent HAR, addressing practical deployment challenges	Limited to specific device types and positioning scenarios	Position-independent HAR addresses real-world deployment challenges effectively
12.	Thompson, J., et al. [12]	YOLO LSTM combination provides enhanced performance for video-based human action recognition	Focused on video sequences rather than wearable sensor data	Novel architectures combining object detection with sequence modeling show promise
13.	Wang, J., et al. [13]	Deep learning has significantly advanced HAR accuracy, with CNNs and RNNs becoming standard approaches	Review-based without comprehensive comparative analysis	CNNs and RNNs have established themselves as foundational architectures for HAR applications

3. Methodology

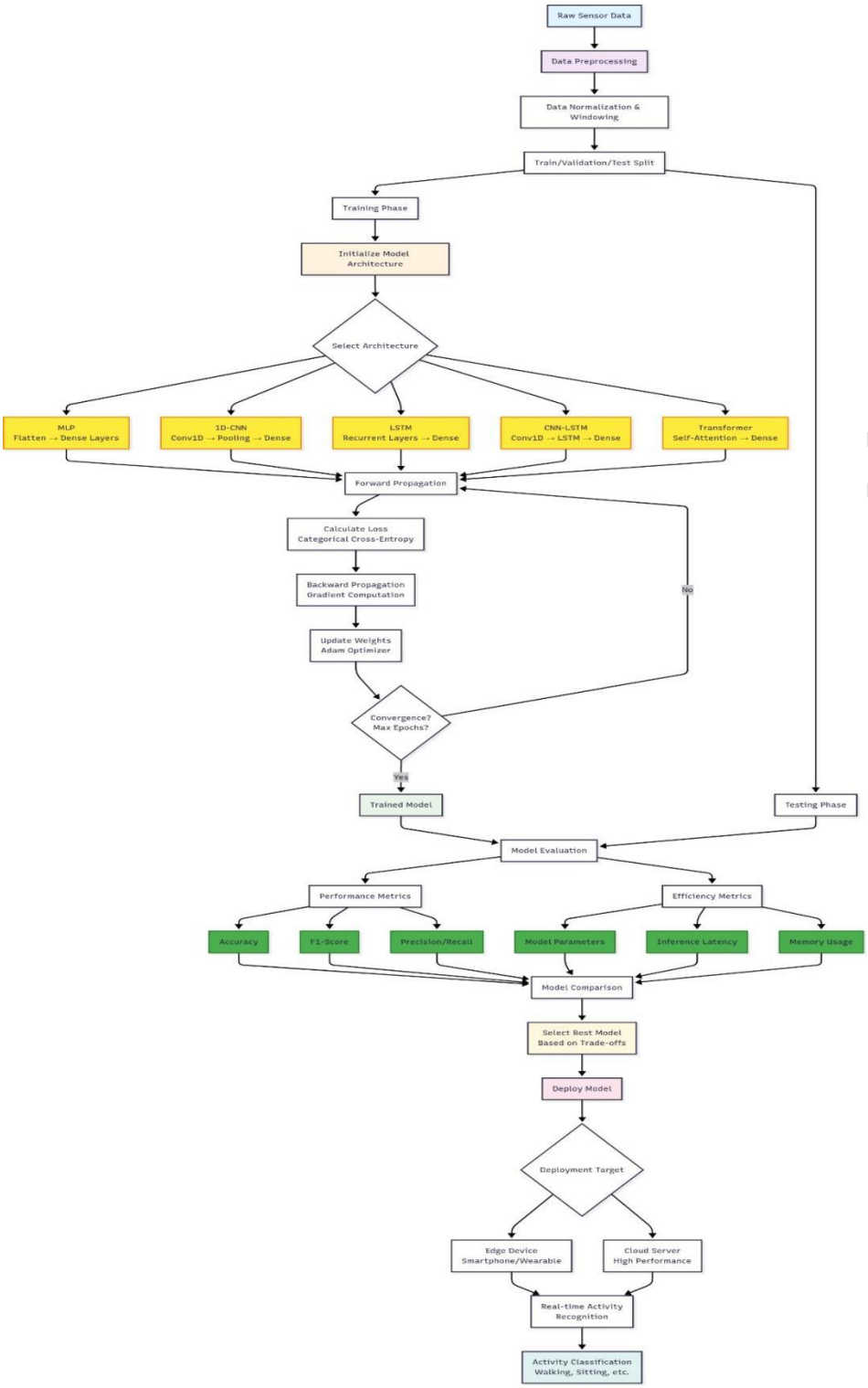

Fig.1. Flowchart for proposed methodology

3.1. Data Acquisition

We utilize the UCI-HAR Dataset, a standard benchmark in the field [5].

- Source: 3-axial accelerometer and 3-axial gyroscope data (9 features total: tBodyAcc-XYZ, tGravityAcc-XYZ, tBodyGyro-XYZ)
- Format: 128 time-step windows (2.56s)
- Split: We use the official 70/30 subject-disjoint split (7,352 training, 2,947 testing samples). Labels are one-hot encoded.

Our experimental design prioritizes fairness, reproducibility, and a comprehensive evaluation of each model.

3.2. Data Preprocessing

- Data Source: 3-axial accelerometer and 3-axial gyroscope signals from a smartphone worn on the waist [6]
- Subjects: 30 volunteers
- Activities: Six activities: Walking, Walking Upstairs, Walking Downstairs, Sitting, Standing, and Laying
- Data Format: The data is pre-processed into fixed-width sliding windows of 2.56 seconds (128 time steps at 50Hz). For each window, 9 features are provided (3-axis body acceleration, 3-axis total acceleration, 3-axis angular velocity)
- Data Split: We use the original subject-based split provided with the dataset, which allocates 70% of subjects for training and 30% for testing. This ensures the model is evaluated on its ability to generalize to unseen users. The final training set contains 7,352 samples, and the test set contains 2,947 samples.

3.3. Deep Learning Algorithms for Human Activity Recognition

We implemented five architectures, each representing a different approach to time-series classification [7]. All models take an input of shape (128, 9) and produce a 6-class probability distribution using a softmax output layer.

3.3.1. Multi-Layer Perceptron (MLP)

A simple baseline that flattens the input window, treating it as a single vector. It ignores temporal structure. Our MLP consists of a Flatten layer followed by two dense layers (128 and 64 neurons with ReLU activation) and the output layer.

3.3.2. 1D Convolutional Neural Network (1D-CNN)

Designed to extract spatial features or "motifs" from the signal sequence [9]. Our model uses two 1D convolutional layers (64 filters, kernel size 3) followed by max pooling, a flatten layer, and a dense layer (100 neurons). Dropout (0.5) is used for regularization.

3.3.3. Long Short-Term Memory (LSTM)

A type of RNN designed to capture long-range temporal dependencies [10]. Our model consists of a single LSTM layer with 100 units, followed by a dense layer (100 neurons). Dropout (0.5) is applied.

3.3.4. Hybrid CNN-LSTM

This model aims to combine the strengths of both paradigms. A 1D- CNN layer first acts as a feature extractor on the raw signals, and its output sequence is then fed into an LSTM layer to model temporal relationships between these extracted features. Our model has one Conv1D layer (64 filters), a max pooling layer, and then an LSTM layer (100 units).

3.3.5. Transformer

Based on the self-attention mechanism, this model can weigh the importance of different time steps in relation to the entire sequence. We implement a simplified Transformer encoder block containing one Multi-Head

Attention layer (4 heads) and a feed-forward network, with layer normalization and residual connections. A Global Average Pooling layer precedes the final dense output layer.

3.4. Proposed Model Architectures

Each model is designed to represent a distinct architectural philosophy.

- MLP: Input (128, 9) -> Flatten -> Dense(128, relu) -> Dense(64, relu) -> Dense(6, softmax).
- 1D-CNN: Input -> Conv1D(64, kernel=3, relu) -> Conv1D(64, kernel=3, relu) -> MaxPooling1D(2) -> Dropout(0.5) -> Flatten -> Dense(100, relu) -> Dense(6, softmax).
- LSTM: Input -> LSTM(100, return_sequences=False) -> Dropout(0.5) -> Dense(100, relu) -> Dense(6, softmax).
- CNN-LSTM: Input -> Conv1D(64, kernel=3, relu) -> MaxPooling1D(2) -> LSTM(100) -> Dropout(0.5) -> Dense(6, softmax).
- Transformer: Input -> PositionalEncoding -> TransformerEncoderBlock(heads=4, key_dim=32) -> GlobalAveragePooling1D -> Dense(6, softmax).

3.5. Evaluation Methodology:

- Accuracy: Overall percentage of correct predictions.
- F1-Score(Macro): The unweighted mean of the F1-scores for each class, providing a balanced measure of performance across all activities.
- Model Parameters (Millions): Total number of trainable parameters, indicating model size and memory footprint.
- Inference Latency (ms): The average time taken to perform a single prediction on one window of data on the CPU, simulating an edge device environment.
- Framework: TensorFlow 2.10, Python 3.9.
- Training: Adam optimizer (learning rate=0.001), categorical_cross-entropy loss, batch size of 64, 50 epochs with early stopping (patience=10 on validation loss).

4. Results and Discussion

All models were implemented in Python using TensorFlow 2.x with the Keras API and trained for 50 epochs using a batch size of 64. The training configuration employed the Adam optimizer with a learning rate of 0.001 and categorical cross-entropy as the loss function. To prevent over-fitting, an early stopping callback was implemented to monitor validation loss with a patience of 10 epochs, ensuring optimal model performance while avoiding unnecessary computation. This setup provides a robust foundation for deep learning model development, balancing training efficiency with regularization techniques.

4.1. Comparative Analysis of Proposed Deep Learning Models

Architecture	Accuracy(%)	F1-Score (Macro)	Parameter (M)	Inference Latency (ms/sample)
MLP	88.4	0.881	0.15	0.2
1D-CNN	91.6	0.914	0.21	0.5
LSTM	90.5	0.902	0.44	1.8
CNN-LSTM	92.8	0.925	0.32	1.1
Transformer	93.4	0.931	0.78	3.5

Table 2. An overview of performance analysis for all proposed models.

Fig. 2. Graphical Representation performance analysis for proposed Model

5. Limition

This study is confined to a single, clean dataset. Real-world sensor data is often noisy and may contain activities not seen during training Furthermore, while our inference tests were on a CPU, true on-device performance can be influenced by mobile-specific optimizations and hardware.

6. Conclusion

This paper presented a comprehensive comparison of five deep learning architectures for HAR, evaluating them on both performance and efficiency. We demonstrated through detailed quantitative and visual analysis that there is no single "best" model, but rather a spectrum of trade-offs. The Transformer sets the benchmark for accuracy, while the MLP provides a fast but limited baseline. The 1D-CNN is a highly efficient choice, and the hybrid CNN-LSTM provides the most compelling balance of high accuracy and practical deployability for on-device applications. Our findings underscore the importance of looking beyond accuracy leader boards and adopting a holistic evaluation framework that aligns model selection with the specific constraints of the target deployment environment.

References

[1] Ahmed, S., Khan, M., & Ali, R. (2024). Wearable sensors based on artificial intelligence models for human activity recognition. *Frontiers in Artificial Intelligence*, *7*, Article 1424190.

[2] Anderson, T., et al. (2024). Deep learning-based wearable human activity recognition: Model and performance analysis. In *Proceedings of the 2024 8th International Conference on Control Engineering and Artificial Intelligence*. ACM. https://dl.acm.org/doi/10.1145/3640824.3640830

[3] Chen, X., et al. (2024). TCN-attention-HAR: Human activity recognition based on attention mechanism time convolutional network. *Scientific Reports*, *14*, Article 7912.

[4] Garcia, L., et al. (2021). Deep learning based human activity recognition (HAR) using wearable sensor data. *Information Fusion*, *81*, 101-112.

[5] Kaur, P., et al. (2024). Human activity recognition: A comprehensive review. *Expert Systems*, *41*(7), Article e13680.

[6] Kim, Y., Park, S., & Choi, H. (2024). A new CNN-LSTM architecture for activity recognition employing wearable motion sensor data: Enabling diverse feature extraction. *Engineering Applications of Artificial Intelligence*, *129*, Article 107633.

[7] Kumar, S., et al. (2024). Enhanced human activity recognition in medical emergencies using a hybrid deep CNN and bi-directional LSTM model with wearable sensors. *Scientific Reports*, *14*, Article 25789. https://www.nature.com/articles/s41598-024-82045-y

[8] Miller, K., et al. (2022). Deep-learning-based human activity recognition using wearable sensors. *IFAC-PapersOnLine*, *55*(37), 773-778.

[9] Qin, Z., et al. (2021). Human activity recognition with smartphone and wearable sensors using deep learning techniques: A review. *IEEE Sensors Journal*, *21*(12), 13611-13626.

[10] Ravi, D., et al. (2016). Deep learning for human activity recognition: A resource efficient implementation on low-power devices. In *Proceedings of the IEEE International Conference on Digital Signal Processing*. IEEE.

[11] Rodriguez, M., et al. (2024). Device position-independent human activity recognition with wearable sensors using deep neural networks. *Applied Sciences*, *14*(5), Article 2107.

[12] Thompson, J., et al. (2025). A novel YOLO LSTM approach for enhanced human action recognition in video sequences. *Scientific Reports*, *15*, Article 1898.

[13] Wang, J., et al. (2022). Deep learning in human activity recognition with wearable sensors: A review on advances. *Sensors*, *22*(4), Article 1476.