# Human Activity Recognition using RNNs

A Preprint

**Lakshay Choudhary**
Masters in Electrical Engineering
University of Stuttgart
Keplerstraße 7, Stuttgart 70174
st176718@stud.uni-stuttgart.de

**Swarnendu Sengupta**
Masters in INFOTECH
University of Stuttgart
Keplerstraße 7, Stuttgart 70174
st180170@stud.uni-stuttgart.de

February 13, 2023

**Abstract**

This paper presents a study aimed at predicting human activities from the Human Activities and Postural Transitions (HAPT) dataset using recurrent neural networks (RNNs). Three different RNN models were evaluated, along with the combination models that incorporated all three. In addition, Bayesian Optimization was done to tune the hyperparameters. Results have shown that one of the combination models outperformed the other models with an accuracy of 94.8%.

## 1 Introduction (Swarnendu Sengupta)

Human Activity Recognition (HAR) has emerged as a crucial area of research due to its vast applications in health monitoring, sports analysis, and elderly care. With the advancement in wearable technology and widespread use of smartphones, an unprecedented amount of data on human movements and activities has become available. In this paper, we present a study aimed at predicting human activities from the Human Activities and Postural Transitions (HAPT) dataset using RNNs.

We evaluated the performance of three different types of RNNs: Long Short-Term Memory (LSTM), Bidirectional LSTM (BiLSTM), and Gated Recurrent Unit (GRU). Additionally, we looked at the effectiveness of combination models that used all three RNNs in every conceivable mix. The outcomes emphasise the value of taking into account both past and future data when predicting human behaviours and show the potential of RNNs in this area. Our findings show the potential of RNNs in this field and a need for further research and also point to the combination model as a potential model for human activity recognition tasks.

## 2 Related Work (Lakshay Choudhary)

HAR has been widely studied in the literature and various approaches have been proposed to tackle this task. Earlier studies in the field of HAR used traditional machine learning algorithms such as decision trees, random forests, and support vector machines (SVMs) for activity recognition. However, the recent advancement in deep learning techniques has led to the development of more sophisticated models, such as convolutional neural networks (CNNs) and RNNs, that have shown promising results.

Several studies have investigated the use of RNNs for human activity recognition. For example, the study in [1] used a combination of LSTM and CNNs to recognize human activities from wearable sensors data. Another study [3] used a BiLSTM network to predict human activities from smartphone accelerometer and gyroscope data.

More recently, the use of Bayesian optimization techniques has been investigated for tuning the hyperparameters of deep learning models in human activity recognition. [2] used Bayesian optimization to tune the hyperparameters of a deep neural network for activity recognition from wearable sensors data.

In this paper, we aim to build upon the existing literature by evaluating the performance of three different types of RNNs, along with a combination model, for HAPT dataset. Additionally, we also investigate the use of Bayesian optimization for tuning the hyperparameters of the models.

## 3 Dataset Description (Swarnendu Sengupta)

The dataset used in this project is derived from the tri-axial accelerometer and gyroscope of a smartphone, which were captured at a frequency of 50 Hz. It has six fundamental (static and dynamic) movements as well as six postural changes between the static movements. While walking, walking up stairs, and lying down are considered dynamic activities, standing, sitting, and lying down are considered static activities. There are six different postural transitions: Stand-to-Sit, Sit-to-Stand, Sit-to-Lie, Lie-to-Sit, and Stand-to-Stand. The postural transitions have less samples, which contributes to an unbalanced dataset, which is crucial to observe. The dataset includes the accompanying activity labels and raw signal data from each participant.

The dataset was divided into train, validation, and test dataset based on the subjects for the purposes of this study. Approximately 70% of the split is for the train dataset (users 01 to 21), 10% is for the validation dataset (users 28 to 30), and 20% is for the test dataset (user-22 to user-27).

## 4 Method and Structure (Lakshay Choudhary and Swarnendu Sengupta)

The aim of this project is to predict human activities based on inertial sensor data collected from smartphones. To achieve this, four different neural network models, including LSTM, GRU, BiLSTM, and six custom models that combine these three layers in different arrangements have been implemented.

The architecture of each of these models is described below and shown in Figure 1 and Figure 2:

**LSTM Model:** This model is composed of two LSTM layers followed by two dropout layers and a dense layer with 13 units and a softmax activation function. The input to this model is a sequence of 6-dimensional feature vectors. The LSTM layers have a specified number of units and return sequences, which allows for the processing of sequential data.

**BiLSTM Model:** This model is similar to the LSTM model, except that the LSTM layers are replaced with bidirectional LSTM layers. The bidirectional LSTM layers process the input sequence in both forward and backward directions to capture the dependencies in the data.

**GRU Model:** This model is similar to the LSTM model, except that the LSTM layers are replaced with GRU layers. The GRU layers are designed to be simpler and computationally efficient than LSTM layers while still capturing sequential dependencies in the data.

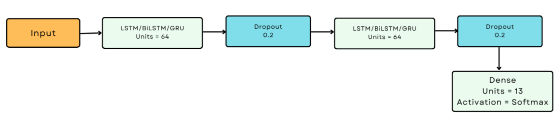



Figure 1: LSTM/GRU/BiLSTM based model

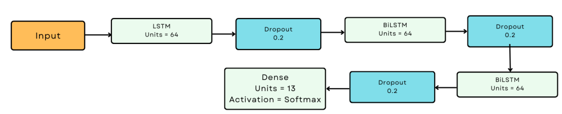



Figure 2: Combination model of all possible combinations of LSTM, GRU and BiLSTM layers

**Combination Models:** We have also implemented six combination models that have LSTM, BiLSTM, and GRU layers in different arrangements. These custom models aim to capture the strengths of the individual models and provide improved performance.

Each of these models includes dropout layers with a specified dropout rate to prevent overfitting. The output of the final dense layer is a 13-dimensional vector, representing the probability of each activity class.

### 4.1 Preprocessing

**Z-Score normalization:** The preprocessing step for this dataset involved normalizing the input data using Z-Score normalization, which is a standard method for scaling the input data to have a mean of zero and a standard deviation of one. This helps to ensure that the input data has a consistent scale and is not dominated by any large outliers or extreme values.

**Sliding Window:** The preprocessing step utilized a Sliding Window technique with a window size of 250 and a shift of 125 (50% overlap). These can be tuned and adjusted. This technique allows the input data to be broken down into overlapping segments, which can be used to capture information about the underlying patterns and trends in the data. By using a 50% overlap, the method ensures that there is some redundancy in the data and that the information captured in each window is not lost when moving to the next.

### 4.2 Implementation Details

**Optimizer:** The implementation of the model involved the use of the Adam optimizer. The Adam optimizer is a popular choice in deep learning due to its ability to effectively adapt the learning rate based on the historical gradient information, which can help to improve the convergence of the optimization process.

**Loss:** The masked sparse categorical cross-entropy loss function was used to calculate the loss for all labels except the unlabeled class. It is based on the cross-entropy loss function, which is defined by equation 1:

$$CE = -\sum_{i=1}^{n} t_i log(p_i) \tag{1}$$

where $t_i$ is the truth label and $p_i$ is the softmax probability for the $i^{th}$ class.

**Metrics:** In evaluating the model's performance, several metrics were used, including confusion metrics, accuracy, F1, precision, and recall. Together, these metrics provide a comprehensive view of the model's performance and can be used to guide further refinement and optimization of the model.

**Training:** The models were trained on 100 epochs with early stopping capability. Early stopping allows the training process to be automatically stopped if the model's performance on the validation set does not improve after a specified number of epochs. This can help to prevent overfitting and improve the generalization of the model. The default learning rate, dropout rate and units used during training were 0.001, 0.2 and 64 respectively.

**Hyperparameter Optimization:** To further optimize the model's performance, Bayesian optimization was performed which uses probabilistic models to guide the search for the optimal set of hyperparameters. Following ranges were given as hyperparameters:

- Units of RNNs' layers ranging from 16 to 256.
- Dropout rate ranging from 0.0 to 0.5.
- Learning rate ranging from 1e-5 to 1e-1.

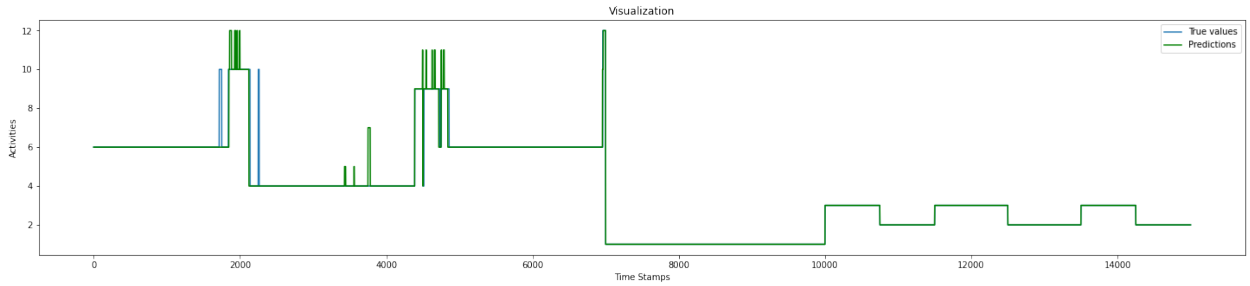
The model was then retrained using the optimal hyperparameters, and the final performance was evaluated using the metrics described in the implementation section.

## 5 Results (Swarnendu Sengupta and Lakshay Choudhary)

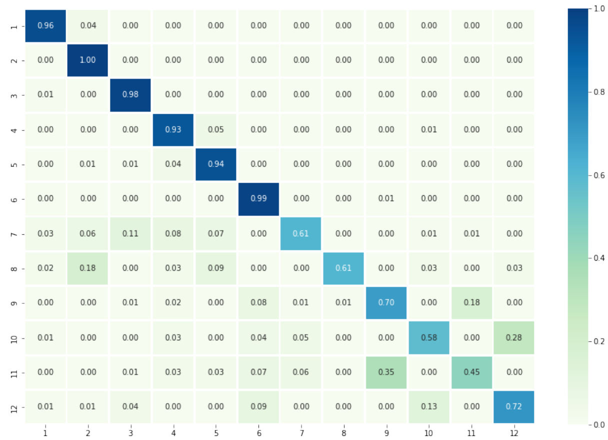
The results on test dataset are shown in Table 1. By means of Bayesian optimization, our best test accuracy is 94.8%. This was generated over 10 runs. The corresponding confusion matrix is shown in Figure 3(b). We randomly choose a segment of sequence in test dataset to visualize the prediction in Figure 3(a).

Table 1: Results on test dataset

| Model along with its description | | | | | |
|---|---|---|---|---|---|
| Model | Description | Dropout | Learning Rate | Units | Test Accuracy |
| LSTM | - | 0.2218 | 0.005421 | 48 | $91.10\,\% \pm 6.34\,\%$ |
| GRU | - | 0.9121 | 0.006895 | 43 | $92.50\,\% \pm 0.63\,\%$ |
| BiLSTM | - | 0.9514 | 0.01 | 51 | $93.94\,\% \pm 0.45\,\%$ |
| Combination Model 1 | LSTM-GRU-BiLSTM | 0.1512 | 0.01468 | 38 | $94.00\,\% \pm 0.96\,\%$ |
| **Combination Model 2** | **LSTM-BiLSTM-GRU** | **0.1512** | **0.01468** | **38** | $94.80\,\% \pm 2.35\,\%$ |
| Combination Model 3 | GRU-LSTM-BiLSTM | 0.1512 | 0.01468 | 38 | $93.94\,\% \pm 1.02\,\%$ |
| Combination Model 4 | GRU-BiLSTM-LSTM | 0.1512 | 0.01468 | 38 | $93.16\,\% \pm 0.35\,\%$ |
| Combination Model 5 | BiLSTM-LSTM-GRU | 0.1512 | 0.01468 | 38 | $93.80\,\% \pm 1.35\,\%$ |
| Combination Model 6 | BiLSTM-GRU-LSTM | 0.1512 | 0.01468 | 38 | $93.00\,\% \pm 0.35\,\%$ |




(a) Visualization of predicted and true activities




(b) Confusion matrix of the Combination model 2

Figure 3: Visualization and confusion matrix of Combination model 2

The result shows that the our network is effective to predict the static activities but not to postural transitions between the static activities. Techniques like upsampling of postural transitions samples can be implemented in the future works.

## 6 Conclusion (Lakshay Choudhary)

The conclusion of this study on human activity recognition using RNNs highlights the robust performance of the Combination model 2. The results showed that this model achieved high accuracy, precision, recall, and F1 scores, which was further improved by the use of Bayesian optimization and early stopping. This demonstrates the potential of the Combination model 2 to be a well-generalized and reliable solution for human activity recognition.

However, it is important to acknowledge the limitations of the model, particularly in recognizing transitional activities between static postures. The results indicate that further work is necessary to address this issue and to enhance the performance of the model in such scenarios.

In conclusion, this study contributes to the field of human activity recognition by demonstrating the potential of using Combination models of RNNs. The results of this study can provide valuable insights for researchers and practitioners in developing practical applications for healthcare, sports, and other related domains.

## References

[1] Aomar Osmani and Massinissa Hamidi. *Bayesian optimization of Neural Architectures for Human Activity Recognition*, pages 171–195. Springer International Publishing, Cham, 2019.

[2] Samundra Deep and Xi Zheng. Hybrid Model Featuring CNN and LSTM Architecture for Human Activity Recognition on Smartphone Sensor Data. In *Proc. of the 20th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT)*, pages 259–264, Gold Coast, QLD, Australia, 2019, IEEE.

[3] Yong Li and Luping Wang. Human Activity Recognition Based on Residual Network and BiLSTM. *Sensors*, 22(2):1424-8220, 2022.