Advancing HAR with GAC Preprocessing and Strategic Sensor Placement in Neural Networks

Menglu Tao†, Diego Zanutti‡

Abstract—**In the rapidly evolving field of sensor-based Human Activity Recognition (HAR), the integration of advanced preprocessing methods and strategic sensor placement has become crucial for maximizing model accuracy and operational efficiency. This report compares two sophisticated preprocessing methodologies—Direct Concatenation Method (DC) and Grouped Activity Concatenation Method (GAC)— and their impact on the performance of diverse neural network architectures, including Dense Neural Networks (DNN), Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks, and Convolutional Long Short-Term Memory (ConvLSTM), within the HAR domain. Our extensive study demonstrates that the CNN and LSTM models outperform the conventional DC technique, with benchmark accuracies near 99.5%, indicating the huge advantage of the GAC in refining model efficacy. Furthermore, through an in-depth analysis dedicated to sensor placement, we determined that putting the accelerometer sensor at the ankle not only increases data accuracy but also considerably improves the system's capacity to recognize complicated actions.These combined findings not only highlight the critical role of cautious preprocessing techniques and thoughtful sensor positioning, but also help to advance the state of the art in HAR systems by demonstrating a methodical approach to achieving unprecedented levels of accuracy and efficiency.**

Index Terms—**Human Activity Recognition, Supervised Learning, Neural Networks, Recurrent Neural Networks.**

I. Introduction

In an era where wearable technology integrates seamlessly into our daily lives, the accurate classification of human activities using these devices has become crucial. This research delves into human activity recognition, a vital area with widespread implications in health monitoring, sports science, and personal fitness, employing neural networks to interpret accelerometry data from multiple body locations.

The innovation of this study is the introduction of a new preprocessing approach and its application to neural networks to improve the accuracy and efficiency of Human Activity Recognition (HAR). Although previous studies have made some progress in processing accelerometer data using traditional machine learning methods, the recognition accuracy of these methods is still insufficient when dealing with complex or similar activities. For example, distinguishing subtle activity differences such as walking and jogging remains a challenge. In addition, traditional methods typically require extensive manual feature engineering, which limits their adaptability to new scenarios and different wear-position data. By introducing an advanced data preprocessing method, this study aims to overcome these limitations and provide a more generalized and automated solution to improve the accuracy and reliability of HAR systems in real-world applications. This new pre-processing method is expected to improve the quality of feature extraction and hence the performance of deep neural networks in recognizing different human activities.

In this paper, we address the question of how to enhance the performance of sensor-based Human Activity Recognition (HAR) systems by improving data preprocessing methods. With the proliferation of smart wearable devices and increased interest in health and fitness monitoring, enhancing the accuracy of HAR systems has become more urgent and important.

The approach we take is to introduce a novel data preprocessing technique that is specifically designed for deep neural network models to improve feature extraction and representation of raw sensor data. The innovation of this approach is that it not only aggregates data from different volunteers, but also groups similar activities using individual and activity IDs, thus avoiding model confusion due to sudden activity changes and effectively improving recognition accuracy.

The value of this study is that it provides a new perspective on the processing of sensor data that may have a profound impact on future research and practice. Compared to traditional methods that rely on extensive manual feature engineering, the automated feature extraction method proposed in this study can process data more efficiently and improve the accuracy and efficiency of the model in complex activity recognition tasks.

In terms of practical applications, the results of this study can be widely used in a variety of scenarios such as health monitoring, sports science, personal fitness, etc., and have potential application value in areas such as smart home and city surveillance. As the method is highly versatile and adaptable, it can be easily integrated into existing products and services, bringing practical benefits to technology companies, research organizations, and even ordinary users. Other researchers and developers can build on this foundation to further explore and extend it, thus promoting the development and application of HAR technology.

The main contributions of this paper are:

Advanced HAR Preprocessing Techniques: We present and compare two complex preprocessing approaches, the Direct Concatenation Method (DC) and the Grouped Activity Concatenation Method (GAC). The latter, a unique preprocessing method, strategically groups data by individuals and activities, considerably lowering model confusion and improving

†Department of Information Engineering, University of Padova, email: {menglu.tao}@studenti.unipd.it

‡Author two affiliation, email: {name.surname}@studenti.unipd.it

recognition accuracy.

Optimized Neural Network Applications: Our study expands the use of deep learning in the field of human activity recognition by conducting a thorough investigation of a variety of neural network architectures, including DNN, CNN, LSTM, and ConvLSTM.The study highlights the outstanding capabilities of CNN and LSTM models, which reached benchmark accuracies of over 99.5%. This demonstrates the efficiency of deep learning in feature extraction and advanced data representation, which considerably improves the adaptability and performance of HAR systems.

Strategic Sensor Placement Insights: A thorough examination into sensor placement has given crucial findings, demonstrating that the ankle is the best site for accelerometer sensors to improve data quality and the system's capacity to reliably recognize complicated actions. This discovery advances the discipline by emphasizing the relevance of sensor placement in boosting the accuracy and efficiency of activity detection systems, ultimately pushing the state of the art in HAR.

This report is organized as follows. In Section II, we present the state of the art (related work), Section III and Section IV describe the processing pipeline, signals and features, respectively. Section V details the learning framework and Section VI exhibits results and their implications, and Section VII performs the concluding remarks.

II. Related Work

In this section, we discuss the research conducted so far in the field of Human Activity Recognition (HAR). The quest for optimal HAR through wearable sensors has spurred diverse methodologies, focusing on data preprocessing, window size optimization, and the exploration of advanced neural network architectures. This section delves into contemporary studies that align with our research objectives, providing a foundation for our innovative approaches to enhance HAR accuracy.

Some of the early research in accelerometer-based activity recognition centered on the use of many accelerometers placed on various regions of the user's body. Bao & Intille [1] collected data from 20 users using five biaxial accelerometers worn on their right hip, dominant wrist, nondominant upper arm, dominant ankle, and nondominant thigh. Acceleration data was collected without researcher supervision or observation. Mean, energy, frequency-domain entropy, and correlation of acceleration data was calculated and several classifiers using these features were tested. Decision tree classifiers showed the best performance recognizing everyday activities with an overall accuracy rate of 84% The results suggest that multiple accelerometers aid in recognition because conjunctions in acceleration feature values can effectively discriminate many activities. However, while multiple sensors provide rich data, they introduce complexity and user discomfort, necessitating a simplified yet effective approach. A study [2] published in PMC explores the extraction of meaningful features from raw accelerometry data to classify walking and its sub-classes. The authors employ Fourier and wavelet transforms for feature extraction and utilize tree-based methodologies for classification, achieving an average accuracy of 87.6% at the subject-specific level. This study not only highlights the impact of sensor location on classification accuracy but also introduces a robust feature inter-subject normalization technique, improving group-level classification accuracy. Their methodology informs our investigation into diverse preprocessing strategies and underscores the potential of tailored feature extraction in enhancing HAR systems.

The impact of window size on activity recognition accuracy is critically analyzed in a comprehensive study [3] referenced in PubMed. This work offers an extensive evaluation of various activity recognition procedures across a range of window sizes, establishing a 1-2 second window as the optimal trade-off for recognition speed and accuracy. This study's insights into the segmentation process and its effect on the activity recognition system's design are instrumental in our examination of choosing the optimal window sizes to optimize our model's performance.

A noteworthy study [4] presents a deep learning network model that classifies various human activities by integrating external features with accelerometer sensor data. Their research underscores the limitations of traditional local features-based approaches and machine learning techniques in capturing temporal information adequately. By experimenting with Long-Short Term Memory (LSTM), Convolutional Neural Networks (CNN), and Convolution Long-Short Term Memory (ConvLSTM) architectures, they demonstrate a significant improvement in algorithm accuracy with the addition of orientation invariant and consecutive point trajectory information to tri-axis accelerometer data. Particularly, the ConvLSTM architecture achieved an accuracy of 98.41% on the WISDOM dataset, surpassing the performance of standalone CNN and LSTM models. While this integration is powerful, it demands a more intricate preprocessing pipeline and higher computational resources. Similarly, the LSTM-CNN hybrid model study [5] presents a sophisticated approach to capturing temporal structures, achieving high accuracy rates on benchmark datasets. Yet, this model's complexity calls for a balance between performance and practicality in real-world applications.

Our contribution to the field addresses these gaps by introducing a novel preprocessing methodology-the Grouped Activity Concatenation Method (GAC)-which simplifies the data preprocessing stage while enhancing model performance. We demonstrate that our approach, when combined with CNN and LSTM models, not only achieves an accuracy of 99.5% but also streamlines the computational process, making it more suitable for real-time applications on resource-constrained devices. Moreover, we extend the knowledge on sensor placement by empirically proving that positioning the accelerometer at the ankle significantly improves data accuracy and the recognition of complex activities.

III. Processing Pipeline

Our study utilizes a dataset provided by PhysioNet containing raw accelerometer data collected from 32 adults during outdoor walking, stair climbing, and driving. The data cover a wide range of ages, genders, and ethnicities, reflecting the diversity of human activity. The data were collected at a frequency of 100 Hz at a total of four body sites, left wrist, left hip, left ankle, and right ankle, capturing the activity of 13 male and 19 female subjects.

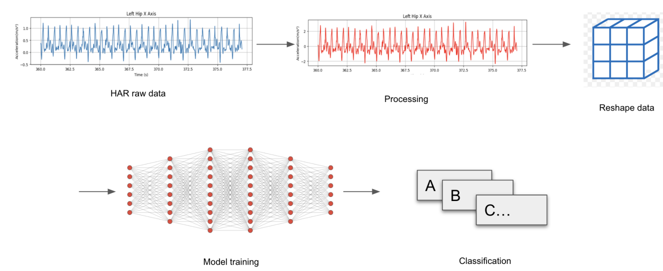

Fig. 1: The data processing pipeline involves loading human activity records, preprocessing the data for machine learning, reshaping it to fit the neural network model, training the model to learn activity patterns, and finally using the trained model for accurate classification of human activities based on sensor data.

In the data processing phase, we first integrated the data records of all volunteers and then grouped them according to individual and activity IDs to ensure accurate delineation of activity types during model learning. Next, the raw data were normalized to standardize the data format and enhance data consistency and reliability. We used a sliding window method to segment the data, which is not only applicable to traditional machine learning but also to the training of deep learning models. This method allows the model to learn time-series features of activities, which improves the accuracy and efficiency of recognition.

A. *Direct Concatenation Method (DC)*

Our initial data processing approach, referred to as the Direct Concatenation Method (DC), involves aggregating data records from all volunteers, and subsequently normalizing the raw data and use a sliding window method to segment the data. This is method is trivial and enables the model to learn time-series features of activities, leading to good accuracy and efficiency of recognition.

B. *Grouped Activity Concatenation Method (GAC)*

The Grouped Activity Concatenation Method (GAC), our refined processing strategy, introduces a novel grouping technique. In GAC, data records are grouped by combining the IDs of the volunteers and the activity type IDs before normalization. This grouping ensures that the model can accurately distinguish the same type of activities performed by different volunteers during the subsequent learning process. For example, the walking data of one volunteer is separated from the walking data of another volunteer, minimizing confusion between the datasets. This grouping strategy is crucial for enhancing the learning efficiency of the model and improving classification accuracy, setting it apart from the DC approach.

Similar to DC, GAC also utilizes the sliding window method for data segmentation. The partitioned data is then used for training, validation, and testing, enabling the model to learn from diverse scenarios and generalize effectively.

By distinguishing between DC and GAC, we aim to highlight the pivotal role of preprocessing methods in shaping model performance and showcase the efficiency gained through our novel Grouped Activity Concatenation approach.

In addition to the processing steps mentioned earlier, we implemented a further method to enhance the robustness of our model by accounting for individual variability in movement patterns. Recognizing that each subject has a unique way of moving and that individuals may exhibit slight variations in their movements, we adopted a strategy to evaluate the model's generalization capabilities across different subjects.

Specifically, we divided the dataset into a training set, which comprised data from 20 randomly chosen subjects, while the remaining subjects' data were reserved for the test set. This approach allowed us to assess the model's ability to generalize and perform effectively on unseen subjects.

This partitioning ensured that the model learned from a representative sample of subjects during training, and the evaluation on a separate set of subjects tested its ability to generalize beyond the specific individuals encountered during training. This methodology serves as a critical step in validating the reliability and applicability of the developed model in real-world scenarios where diverse individuals may be encountered.

IV. Signals and Features

The accelerometers used in the dataset collected motion data at four different body sites, recorded as fixed-size vectors. This formatted data allowed us to capture and analyze activity patterns at regular intervals, providing a solid foundation for subsequent feature extraction and model training.

To improve the quality of the data, we normalize the signal to ensure that it is stable. These preprocessing steps are crucial for subsequent feature extraction and model training, and they ensure clean and consistent data.

After preprocessing the signal, we use a method called "sliding window" to segment the data. This method is applicable to both classical machine learning methods based on traditional manual feature extraction and neural networks. It divides the input data signal into multiple signal windows, each covering a certain range of observation times (e.g., 2.56 seconds or 5.12 seconds), with different lengths of the windows affecting the training results of the final model. In each signal window, there are a certain number of data points, each of which has a different feature value (variable) as well as a corresponding type of activity.

In addition, we used the strategy of overlapping windows when splitting the windows, and the overlap rate was set according to the actual situation, in this study we set it to 50%, so that the first half of the window contains observations from the second half of the previous window. This approach ultimately produces a three-dimensional matrix [samples, time

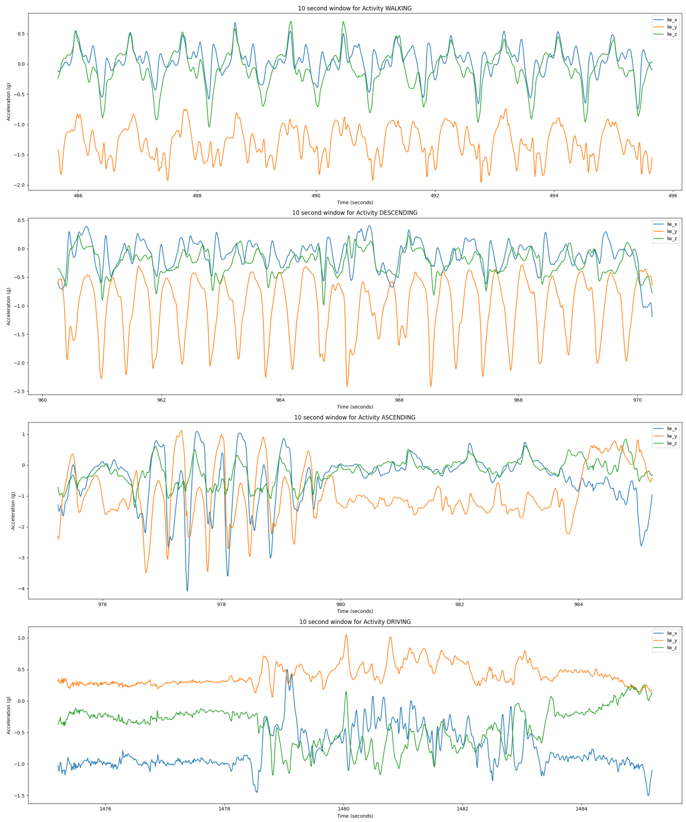

Fig. 2: Acceleration data of the left wrist sensor for different activities, the dataset have a sampling frequency of 100 Hz, the window duration in this image is of 10 seconds. Each axis is shown in a different color: lw_x in blue, lw_y in orange, and lw_z in green.

steps, features] that serves as the input format for the neural network.

After processing multiple data segments, we partitioned them into three datasets, training, validation, and testing, for subsequent model training and evaluation.

V. Learning Framework

When working with the PhysioNet dataset, we first used a special grouping strategy to group the data by combining the IDs of the volunteers and the activity type IDs. The purpose of this step is to ensure that the model can accurately distinguish the same type of activities performed by different volunteers during the subsequent learning process. For example, the walking data of the same volunteer is separated from the walking data of another volunteer as a way to minimize the confusion between the data. This grouping strategy is crucial for enhancing the learning efficiency of the model and improving the accuracy of the classification.

With a batch size of 32 samples, the training is conducted using a mini-batch technique. For a total of five epochs, trainings were conducted again in order to provide a recurrent learning process and a more accurate learning. Training is done across the whole training set, and to verify the performance of the generated model, loss and accuracy are checked after every training set, including the validation set. Additionally, the ”Adam” optimizer has been employed to carry out the learning process. This Keras built-in approach arranges the learning rate dynamically and differently for each parameter, enabling a faster and more accurate learning process.

A. *Deep Neural Network*

The first neural network is dense and fully connected. The architecture consists of three dense layers with 256, 128, and 64 neurons each, followed by a dropout layer with a rate of 0.3 to prevent overfitting. Each thick layer additionally receives L2 regularization, which penalizes excessive weights and helps to reduce overfitting.The L2 regularization term is added to the loss function to penalize large weights, and is defined as:

$$L2 = \lambda \sum_{i=1}^{n} w_i^2$$

where λ is the regularization coefficient (in this case, 0.001) and w_i represents the weights of the neural network. The model uses ReLU activation functions for hidden layers to introduce nonlinearity, allowing it to learn complicated patterns from activity data. The input data is intended to have a dimensionality of 1536 (128*12) to accommodate features retrieved from time-series sensors. The network culminates in a softmax output layer with four units representing the four activity classes, which provide a probability distribution over these classes. The model employs the Adam optimizer with a learning rate of 0.001 and is trained with a loss function of sparse categorical crossentropy to improve classification accuracy.

B. *Convolutional Neural Network*

This Convolutional Neural Network (CNN) model begins with a Conv1D layer with 64 filters and a kernel size of 3, capable of catching local patterns within the input stream. This is followed by a MaxPooling1D layer, which cuts the sequence length in half and improves the model's capacity to focus on the most important features by downsampling the feature maps.

The use of Dropout layers after each pooling process, with a dropout rate of 0.25 to reduce overfitting, indicates a careful approach to model generalization. Subsequent convolutional layers, with filter counts of 128 and 256, show a deepening network extracting increasingly abstract properties from the input data. These layers are supplemented by max pooling and dropout layers, which help the model focus on important features while preventing overfitting.The model uses the Flatten operation to shift from convolutional to dense layers, altering the final result to fit dense layers. A dense layer with 128 cells follows, introducing a high-level reasoning phase over the extracted features, before concluding with a final dense layer of four neurons using a softmax activation function. This output layer shows that the model's objective is to classify input sequences into one of four categories.

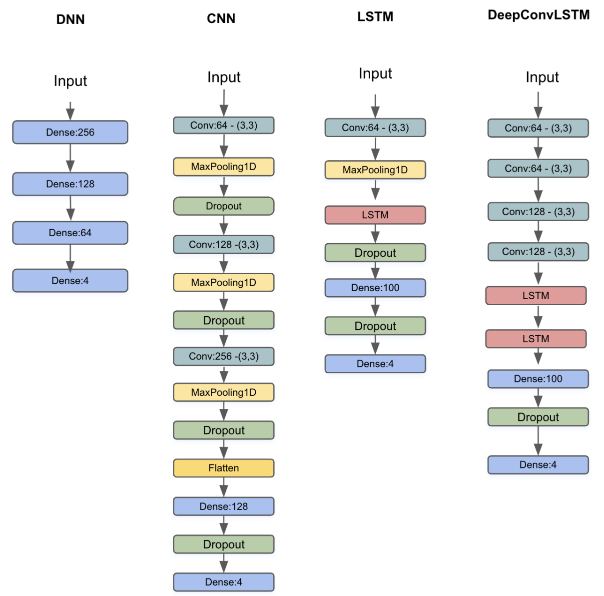

Fig. 3: Model Structures

C. *Long short-term memory Network*

The third model is the LSTM model, which begins with a Conv1D layer capable of recognizing local patterns in the input sequence. This layer, which has 64 filters and a kernel size of three, efficiently captures key properties inherent in sequential data. The sequence is then downsampled using a MaxPooling1D layer, accentuating important features while minimizing computational complexity. Following the convolutional step, the model shifts to recurrent processing by introducing an LSTM (Long Short-Term Memory) layer with 32 memory units. This LSTM layer excels at capturing temporal dependencies in sequential data, allowing the model to recognize detailed patterns across the input sequence. To reduce overfitting and improve model generalization, a dropout layer with a rate of 0.3 is strategically placed after the LSTM layer. The recurrent output is then routed into a dense processing phase, which begins with a fully linked dense layer of 100 neurons activated by the rectified linear unit (ReLU) activation function. This dense layer allows for high-level synthesis of the learnt features, which enhances the model's representation of the input sequence. To prevent overfitting and improve model robustness, an additional dropout layer with a rate of 0.3 is used. The final layer of the network is a dense layer with a softmax activation function, which serves as the model's output layer. The number of neurons equals the number of classes.

D. *DeepConvLSTM*

The Deep Convolutional LSTM model is a complex architecture that starts with a succession of Conv1D layers that are purposefully intended to extract hierarchical features from input sequence data. Each Conv1D layer uses a set of filters (64 and 128 respectively), each with a kernel size of three, and is activated using the rectified linear unit (ReLU activation function.Each Conv1D layer uses a set of filters (64 and 128 respectively), each with a kernel size of three, and is activated using the rectified linear unit (ReLU) activation function.Following the convolutional stage, the model easily shifts to recurrent processing with the addition of two LSTM (Long Short-Term Memory) layers. The first LSTM layer, which has 100 memory units and is programmed to return sequences, helps to identify temporal dependencies and nuanced patterns from convolved information. The succeeding LSTM layer, which also has 100 memory units, refines the learnt representations and captures the sequential dynamics of the input data. The recurrent output is then routed through a dense processing step, which begins with a fully connected dense layer of 100 neurons driven by the ReLU activation function. This dense layer promotes a high-level synthesis of the learnt features, which improves the model's ability to identify intricate patterns and relationships in the data. To avoid overfitting and ensure model generalization, a dropout layer with a 0.5 dropout rate is carefully inserted.The output layer of the network consists of a dense layer with a softmax activation function. This layer, which has the same number of neurons as classes, orchestrates the classification process by allocating probability distributions across the various class labels. The Adam optimizer is used to train the model with a specific learning rate, optimize the sparse categorical cross-entropy loss function, and evaluate its performance using accuracy measures.

VI. Results

This section presents the outcomes of the trained models and experimental evaluations. The performance of different network architectures, such as DNN, CNN, LSTM, and Conv+LSTM, is thoroughly evaluated.

Additionally, the effect of a novel preprocessing strategy on model performance is investigated. Furthermore, the findings of analyzing data from various body positions are reviewed to establish the best location for the accelerometer sensor for activity identification.

Architecture	Accuracy	Precision	Recall	F1	Loss
DNN	96.2%	96.1%	96.2%	96.1%	3.60%
CNN	99.5%	99.5%	99.5%	99.5%	1.93%
LSTM	99.5%	99.5%	99.5%	99.5%	2.06%
DeepConvLSTM	98.4%	98.4%	98.4%	98.3%	5.31%

TABLE 1: Performance Comparison of Different Network Architectures

Table 1 compares performance results for various models, including DNN, CNN, LSTM, and DeepConvLSTM. The DNN model had an accuracy of 96.2% and an F1 score of 96.1%. CNN and LSTM outperformed with an accuracy of 99.5% and an F1 score of 99.5%, suggesting its ability to detect complex patterns in data. The DeepConvLSTM model also performed admirably, with an accuracy of 98.4% and an F1 score of 98.4%. These findings demonstrate the efficiency

	DC		GAC	
Architecture	**Accuracy**	**F1**	**Accuracy**	**F1**
DNN	89.3%	89.0%	96.2%	96.1%
CNN	89.5%	90.5%	99.5%	99.5%
LSTM	93.3%	93.7%	99.5%	99.5%
DeepConvLSTM	90.7%	91.6%	98.4%	98.4%

TABLE 2: Comparison of preprocessing methods: The first two columns present results obtained using the Direct Concatenation Method, while the last two columns showcase results from the Grouped Activity Concatenation Method

of the suggested models in accurately identifying human activities using accelerometer data.

Additionally, to provide further insights, confusion matrix and training records fig. 4 and fig.5 for the CNN and LSTM models, showcasing their classification performance, are included below. The confusion matrix shows that the models are doing an excellent job discriminating between driving and ascending stairs. However, when it comes to distinguishing between walking and descending steps, the models make more mistakes.

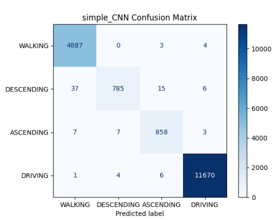

Fig. 4: Confusion Matrix for CNN Model

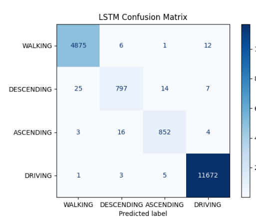

Fig. 5: Confusion Matrix for LSTM Model

Table 2 compares the performance metrics for various preprocessing methods. The left two columns show the results of the Direct Concatenation Method, while the right two columns show the results of the Grouped Activity Concatenation Method. As demonstrated, the latter strategy produces considerable benefits across a variety of criteria. Notably, both accuracy and F1-score improve significantly, demonstrating the efficiency of the Grouped Activity Concatenation Method in enhancing model performance for human activity recognition. These findings emphasize the relevance of preprocessing strategies in maximizing model outcomes for difficult tasks like activity recognition.

The graphic Fig. 6 demonstrates the accuracy achieved by several networks across multiple body positions, specifically the wrist, hip, and ankle. Each bar represents a network, and the grouped bars show the accuracy of each body position.

Upon closer study, a similar trend emerges: accuracy is highest when the accelerometer sensor is placed at the ankle position, as opposed to the wrist and hip positions across all networks. This shows that the ankle position could provide more useful information for accurate activity detection.

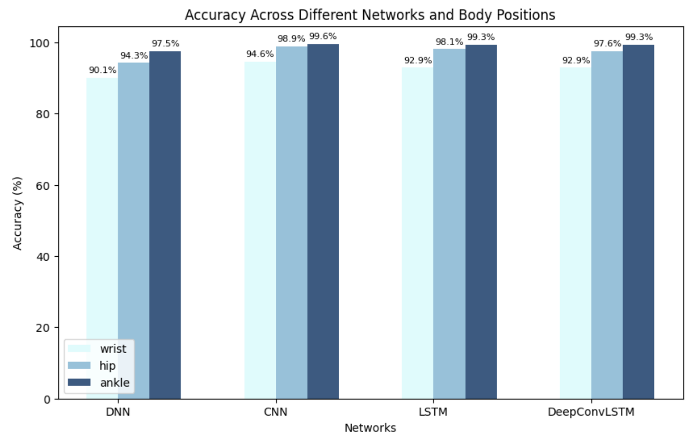

Fig. 6: Accuracy Across Different Networks and Body Positions, we can see that accuracy is highest when the accelerometer sensor is placed at the ankle position.

VII. Concluding Remarks

The present study investigates several elements of sensor-based Human Activity Recognition (HAR), emphasizing the importance of improved preprocessing approaches and sensor location in improving model accuracy and efficiency. We exhibited an important boost in model performance by introducing and comparing the Direct Concatenation Method (DC) and the Grouped Activity Concatenation Method (GAC), with the GAC particularly reducing model confusion and increasing identification accuracy. Our investigation into the use of deep neural networks,specifically DNN, CNN, LSTM, and ConvLSTM, uncovered the powerful capabilities of CNN and LSTM models, which achieved accuracies of up to 99.5%. Furthermore, our sensor placement research identified the ankle as the best location for accelerometer sensors, a discovery that significantly enhances data quality and activity detection precision.

In the context of deploying the proposed Human Activity Recognition (HAR) system in online applications, especially on mobile devices with limited computing power, we made strategic decisions to streamline the architecture. Batch normalization was initially used to deal with overfitting, and although effective for improving training stability in deep neural networks and providing good performances, was omitted in our design. Given the hypotetical application to mobile devices, we prioritized simplicity and computational efficiency. Overfitting concerns were already addressed through the implementation of dropout layers during model training.

Moreover, during the feature selection phase, we initially explored the potential benefits of incorporating vector magnitude and displacement vector as input features. However, after comprehensive experimentation, we decided to discard these features and instead rely solely on raw acceleration data. This decision was motivated by the remarkable performance achieved through the streamlined approach of cleaning raw data. The exclusion of vector magnitude and displacement vector not only simplified the model but also underscored the robustness of our preprocessing methods. The model's capacity to attain high accuracy without relying on additional features highlights the efficacy of our refined approach.

Additionally, the removal of vector magnitude and displacement vector was motivated by the desire to optimize the model for real-world scenarios. By focusing on the essential information provided by raw acceleration data, our system becomes more adaptable to diverse contexts, ensuring reliable performance across different activities and user scenarios. This strategic simplification aligns with the practical considerations of deploying HAR systems in resource-constrained environments.

REFERENCES

[1] Ling Bao and Stephen S. Intille. Activity recognition from user-annotated acceleration data. In *Pervasive Computing. Pervasive 2004. Lecture Notes in Computer Science*, volume 3001, pages 1–17. Springer, Berlin, Heidelberg, 2004.

[2] Wael Fadel, Jakub K. Urbanek, Steven R. Albertson, Xiao Li, Andrea K. Chomistek, and Jaroslaw Harezlak. Differentiating between walking and stair climbing using raw accelerometry data. *Stat Biosci*, 11(2):334–354, 2019.

[3] Oresti Banos, Jose M. Galvez, Miguel Damas, Hector Pomares, and Ignacio Rojas. Window size impact in human activity recognition. *Sensors (Basel)*, 14(4):6474–99, 2014.

[4] Neha Varshney, Bhishamvar Bakariya, A. K. S. Kushwaha, et al. Human activity recognition by combining external features with accelerometer sensor data using deep learning network model. *Multimed Tools Appl*, 81:34633–34652, 2022.

[5] Lingjuan Lyu, Xuanli He, Yee Wei Law, and Marimuthu Palaniswami. Privacy-preserving collaborative deep learning with application to human activity recognition. In *CIKM '17*, 2017.