A Comparative Study of LSTM-based Multimodal Human Activities Recognition

Zeqi Zhong
Electronic and Computer Engineering, Department of Electronic, Electrical and Systems Engineering, University of Birmingham

***Abstract*—The aim of this paper is to investigate the impact of incorporating gravity and attitude data into gyroscopes- and accelerometers-based datasets on the performance of multimodal human activities recognition. Most human activities recognition studies have been based on two sensor data, gyroscopes and accelerometers. However, with the increasing number of smart devices that can now host multiple sensors, there are many potential possibilities that can enhance human activity recognition performance. In order to understand the impact of other modalities on human activity recognition, this paper trains and evaluates Long Short-Term Memory (LSTM) models for different modalities through comparative experiments, with the condition of equal weights for all modalities, and records the evaluation process in detail. The results were evaluated in two ways: confusion matrix and global interpretation of SHapley Additive exPlanations (SHAP) values. The experimental results show that the model using all modalities outperforms the other models in terms of recall and precision, even though the model performance does not increase linearly with the number of modalities. On the other hand, after a global interpretation of the anomalous result "walking", it was found that gravity causes the model to misclassify "walking" as "jogging" more easily, but gravity and attitude data together can enhance the contribution of gyroscope and acceleration data in the model. From the results, using all 4 modalities gives the best performance of the model.**

***Index Terms*—Deep Learning, Multimodal, Human Activities Recognition**

I. Introduction

Humans understand the world in a multifaceted way, and humans can build up a perception of things through a variety of senses, which can be thought of as the modalities that make up human cognition [1]. In the field of artificial intelligence, in order for a model to have the same understanding as a human, the model needs to be able to understand more than one modality [1], and more than one modality is also known as multimodal. A key concept in the use of multimodal data in AI models is the "complementarity" [2] of multimodal data. In multimodal data, different modalities may provide different types of information that together can provide a complete message. By using multimodal data, complex tasks can be learnt in a complementary way [2]. Human activities, as a class of very complex activities, fit very well into application scenarios that use multimodal data. It is quite normal for AI models to use multimodal data in recognising human activities [2], known as multimodal HAR (Human Activity Recognition).

Deep learning, as an approach that has revolutionised traditional machine learning within the last decade, has also performed well in the field of HAR [3]. One of the big advantages of deep learning over traditional machine learning methods is that deep neural networks can efficiently learn features in raw data without manually designing and extracting the feature [3]. In the case of multimodal HAR, the complex and specialised step of extracting features, which requires prior knowledge in traditional multimodal learning, can be eliminated by using deep learning [2]. And, compared to the relatively small datasets used for traditional multimodal recognition, multimodal deep learning allows the use of larger and more comprehensive raw datasets for training [2].

Multimodal HAR based on deep learning, on the other hand, well integrates the advantages of deep learning, multimodality, and becomes a very efficient approach to do HAR.

The development of HRA is inextricably linked to the development of wearable or mobile smart devices, which have greatly stimulated the development of HRA [3]. These devices usually integrate a variety of sensors that monitor a wide range of human physiological signals and perform HAR, providing better service in many fields like healthcare, entertainment, and industry [3]. The market potential for wearables, which are particularly relevant to HAR, is also huge, with the market for these devices predicted to reach almost $63 billion by 2025 [3], which also means that HAR applications are also very promising commercially.

With the development of smart devices, a smart device can easily integrate multiple sensors. For mobile phones, the vast majority of mobile phones can now easily integrate more than 10 various sensors [2]. This also brings a new problem for multimodal HAR: how to combine different kinds of sensor data and provide customers with smarter services to enhance user experience?

Ferrari et al. explored the performance of gyroscopes and accelerometers in multimodal HAR in their 2019 study [4]. They divided the HAR data into 3 groups: gyroscope alone, accelerometer alone and both together. Depending on the grouping, training and evaluation were performed in the non-deep machine learning models KNN, SVM and the deep machine learning model ResNet. Moreover, the two non-deep machine learning models use four different feature extraction methods respectively. The results show that in terms of model performance, the deep learning model ResNet leads the KNN and SVM models in terms of accuracy in recognition across the board. In terms of grouping, models that use gyroscopes and acceleration together are more accurate than models that

use either type of data alone.

This study will build on the research of Ferrari et al., to investigate the effects of gravity data and attitude data on accelerometer- and gyroscope-based data. The dataset used in this study is from the data anonymisation study by Malekzadeh et al [5]. This research can explore how the recognition performance of HAR can be improved by adding effective sensor data to provide a better user experience in advanced and more sensor-rich devices.

In this study, the control variable method was used to divide the dataset into several control groups in order to investigate the effect of two modalities, gravity and attitude, on accelerometer- and gyroscope-based modal data. After selecting and designing a suitable deep learning neural network, the model will be trained and evaluated according to the grouping information. The results of the evaluation will be recorded in detail in the confusion matrix for error analysis. In addition to this, this study will also provide a global interpretation of some anomalous results to resolve the specific behaviour of the two modalities, gravity and attitude, on the HAR in more depth.

This report will first describe the format of the data, the methods for detecting and reducing data noise, and grouping. Then several candidate neural networks will be compared, and a suitable network structure will be designed based on the final choice. How to implement the network in code will also be mentioned. Finally, the report will be analysed in detail based on the evaluation results to get the final conclusion.

II. Data Pre-processing

For traditional machine learning algorithms, datasets usually need to extract features based on expertise before being imported into the model, in order to get better results [3] . However, for neural networks, a deep machine learning method that is particularly sensitive to raw data [3], complex pre-processing of raw data for feature extraction may not be the best approach. Even though with no feature extraction pre-processing, noise reduction of the original dataset is completely necessary to obtain a good quality dataset while preserving the features of the original data [6]. This section will start with an introduction to the structure of the dataset and select the appropriate noise detection and noise reduction methods based on a detailed analysis of the data.

A. Structure of Data

The dataset used in the study contains multiple feature fields. This dataset was collected from the Core Motion framework for IOS devices [5] and the data read from CMDeviceMotion class provide three-dimensional data for attitude, rotation rate, gravity and acceleration [7], a total of 12 feature data that allows for human activity recognition.

With the addition of the index numbers and the corresponding labels, the complete format of the data used for training as well as evaluation is shown in Table I. Each row of data has 14 elements, where the beginning is the index number, the end is the label and the middle is the valid feature data.

The labels at the end of the data structure contain a total of 6 different human activities: downstairs (DWS), upstairs (UPS), walking (WLK), jogging (JOG), sitting (SIT), and standing (STD).

TABLE I
Standard structure for a one-row dataset

Index	***Column Name***	***Data Content***	***Feature Type****
1	Unnamed: 0	Index number	
2	attitude.roll	Roll attitude	ATT
3	attitude.pitch	Pitch attitude	ATT
4	attitude.yaw	Yaw attitude	ATT
5	gravity.x	x axis gravity	GRA
6	gravity.y	y axis gravity	GRA
7	gravity.z	z axis gravity	GRA
8	rotationRate.x	x axis rotation rate	GYRO
9	rotationRate.y	y axis rotation rate	GYRO
10	rotationRate.z	z axis rotation rate	GYRO
11	userAcceleration.x	x axis acceleration	ACC
12	userAcceleration.y	y axis acceleration	ACC
13	userAcceleration.z	z axis acceleration	ACC
14	Label	Label	

*ATT: Attitude, GRA: Gravity, GYRO: Gyroscope, ACC: Accelerometer.

B. Data Grouping for Multimodal Recognition

In this study, control variables will be used to group the datasets for the purpose of multimodal recognition. Ferrari et al.'s multimodal recognition research [4] demonstrated that the combined use of accelerometers and gyroscopes can have better recognition performance, which is better than using either sensor alone.

Therefore, the basis of grouping for this study will be gyroscope plus accelerometer data, and no deletion will be made to the six variables for these two features. The study will be based on these two features, investigate the effect of attitude and gravity data on human activity recognition. According to this idea, the dataset will be divided into 4 groups, use for training and testing:

- Group 1: ACC+GYRO
- Group 2: ACC+GYRO+GRA
- Group 3: ACC+GYRO+ATT
- Group 4: ACC+GYRO+GRA+ATT(ALL)

Features that are not included in the group will be uniformly assigned an invalid value of 0. This way the model may not be able to use this feature as a basis for classification judgement. In this approach, there is no need to change the structure of the neural network, which means that there is no need to change the 12 input features when training and testing the model for different groupings.

Moreover, the weights between the 4 different modalities are equal during training and evaluation which is also a part of the control variables.

C. Noise Detection

Noise is an outlier to a dataset that does not match the expected behaviour, and these data are usually unwanted [6]. Typically, eliminating this noise can better aid the model

in making predictions [6], and detecting data outliers is a necessary and important precursor to performing data noise reduction.

In the context of a normally distributed dataset, approximately 99.7% of the data falls within three standard deviations. If 3 times standard deviation is used as the noise judgement condition for this non-normally distributed HAR dataset, the results of the detection are shown in Figure 1.

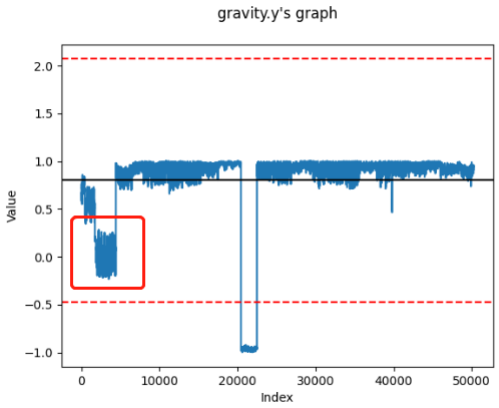


Fig. 1. Noise detection results using 3x standard deviation. The red dotted line is the noise judgement condition and the red box is obvious noise.

Obviously, the detection results of this method are not satisfactory, with visually apparently observable noise contained within the range of the normal signal, shown in the red box of Figure 1.

An outlier detection strategy that is more suitable for dealing with non-normal distributions is Tukey's Fences [8], which sets a normal signal range of

$$[(Q1 - K * IQR), (Q3 + K * IQR)] \tag{1}$$

where K is an adjustable parameter, practically set at 2. Q1 is the first quartile and Q3 is the third quartile. IQR (interquartile range) is (Q3-Q1).

From the results, Tukey's Fences detection approach is good and reasonable, as shown in Figure 2. Therefore, this approach will be used for the entire dataset to perform noise detection for noise reduction in the next step.

D. Noise Reduction

In case of noise happening in the dataset, these outliers should be removed or corrected to obtain a higher-quality dataset [6]. Unlike traditional machine learning methods, neural networks perform better in learning the original data [3], so the aim of this noise reduction is to minimise the impact of outliers while maintaining the original distribution of the data.

The selection of noise reduction methods needs to be cautious, and some methods may affect the raw distribution of the data. For example, removing outliers is a controversial practice [8], especially for the time-series dataset used in this study, where the data are time-dependent.

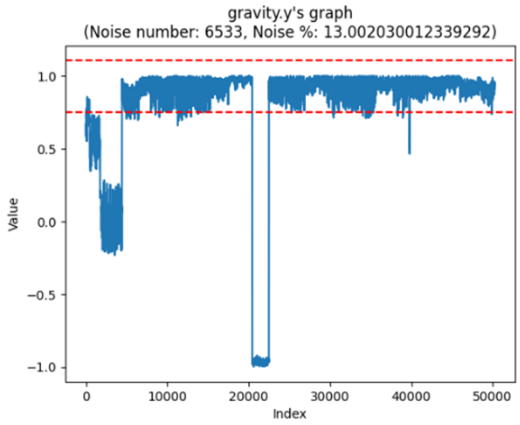


Fig. 2. Noise detection results using Tukey's Fences. The red dotted line is the noise judgement condition.

In order to select the noise reduction method, it is necessary to analyse the dataset. By picturing the dataset and analysing each feature variable, the specifics of the dataset are:

- Visually, noisy data is moved up and down by a shifting noise, and the distribution after it has moved contains valid information (This is confirmed by comparing data with similar individual information).
- Each of the 12 variables in the dataset identifies a different amount of noise.
- A portion of the noise is continuous in time series.

Due to the inconsistent amount of noise detected for each different feature variable, removing outliers could result in cluttering the data and affecting the distribution of the original data. This is contrary to the goal of noise reduction.

Another common approach is to replace the outliers with mean, median or boundary values. However, for continuously occurring noise, the result of noise reduction will make the part of dataset appear as a straight line. This also violates the noise reduction goal of not affecting the original data distribution.

Ultimately, based on the first property observed, shifting will be used as noise reduction for the dataset.

The noise reduction program will loop through the filter several times, each time only shifting the numbers that are outside the normal range of values. Once the data reaches the normal range of values through shift, no more shift processing will be performed.

For the basic step unit that shift adjusts each time, Use the IQR of Equation (1) as the base step size and adjust the length of the step with K*IQR. This parameter is the difference between the third quartile and the first quartile for each variable. Compared to a fixed step size, IQR is more adaptable to all feature variables.

In practice, when the step size is 5*IQR and after 10 iterations, the effect of noise reduction is relatively good, as shown in Figure 3.

There are two reasons for setting such parameters:

- If the step size is too small (like 2*IQR), it may result in the noise moving to the boundary with some of the

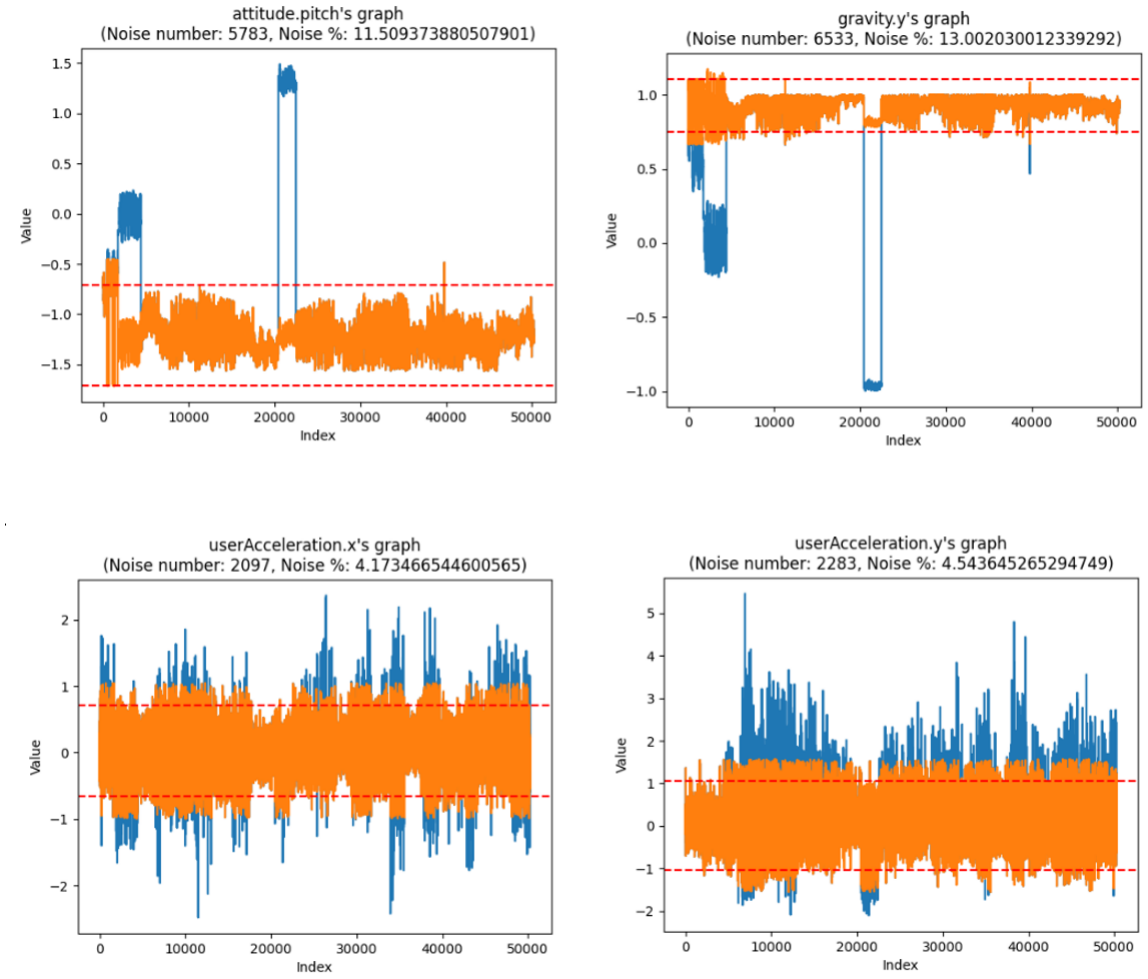


Fig. 3. Some noise reduction results using the Shift method. The blue portion is the original line graph and the orange portion is the line graph after noise reduction.

data inside the boundary and some outside. Shifting again would then destroy the original distribution of the data.

- With each iteration, the data may shift slightly. If there are too many iterations, the same situation of being stuck at the boundary will occur.

Therefore, a relatively medium step size as well as a shorter number of iterations can have better noise reduction.

III. Network and Software Programming

Neural networks, as the core part of human activity recognition, and the analysis of the outputs of neural network are the key to carry out the multimodal recognition study. Therefore, it is important to choose a network structure that is suitable for this multimodal recognition study. By analysing this experiment, the following key points are:

- The recognition accuracies in this study are only intended to compare the recognition results of datasets with different features for multimodal recognition studies.
- The dataset for this experiment is time-series.

According to these, the neural network requirement for this research is: a network that can be trained and tested on time-series data, and its recognition performance is not the main consideration, as long as it is reasonable. For these reasons, LSTM was chosen as the main body of the network.

This section will first introduce the reasons for using LSTM and then show the structure of the entire neural network. Finally, it will show a portion of the key software code writing flow.

A. Network Selection

Depending on the requirements of the research on the network, there are two suitable and common network structures: Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN).

Typical RNN has the ability to include previous states as part of the current state [9], which gives the RNN the ability to deal with datasets with dependencies, such as time-series datasets. A major problem with RNN is that it suffers from gradient vanishing or explosion when dealing with datasets with

long-term dependencies. However, Long Short-Term Memory (LSTM), a variant of RNN, was first proposed in 1997 by Hochreiter and Schmidhuber [10], can solve the problem of long-term dependence to alleviate the gradient issue [11].

Another option CNN is very common in visual recognition. A common solution for CNN where the gradient explodes or disappears is ResNet [12]. Normally CNN cannot handle temporal input data [9], but by modifying it and turning the CNN into a 1-dimensional convolutional neural network, it has the ability to handle temporal data, such as in Ferrari et al.'s multimodal recognition research [4], uses a 1-dimensional ResNet as the network for recognition.

Based on the characteristics of these neural networks, the candidate network structures are LSTM and ResNet. Comparing these two network structures, it can be seen that both have the ability to process time-series data, and both of them can alleviate the problem of vanishing or exploding gradients.

However, CNN is more adept at handling data with static features, such as images; in contrast, RNN is more suited to handling data that is sequential, where the order of this data is important to the overall dataset [13]. The dataset used in this research is more in line with the type of data that RNN is good at handling, so LSTM was chosen as the main body of neural network for this research.

B. Network Structure

Under the condition of using LSTM as the core part of the network, some other layers are needed to make this neural network with 6 classification (DWS, JOG, SIT, STD, UPS, WLK) functions. Figure 4 shows the complete structure of the network, demonstrating how to go from 12 feature inputs to 6 classification result outputs.

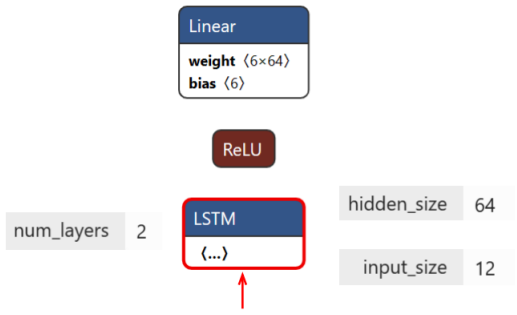


Fig. 4. Network structure parsing results from the Netron website (https://netron.app/). The red arrow shows the feedforward direction of the network.

The model starts with a 12-feature input layer into two LSTM hidden layers, each LSTM layer with 64 hidden units.

Next, the output of the LSTM layer passes through an activation function layer, here the ReLU (Rectified Linear Unit) function is chosen [14], which is a commonly used nonlinear activation function.

Finally, the output of the ReLU layer is processed through a fully connected layer which has 6 neurons corresponding to 6 different human activity categories. For the 6 outputs, Pytorch's torch.max() method [15] is used to find the category with the highest probability and this max value is used as the final recognition result of the model.

C. Software Programming

The neural networks used for training and testing in this study are done through Python and use the PyTorch neural network framework. A complete training process needs to include three main code parts:

- Part 1. Structure of the neural network. This part defines a forward method that determines how the input data is propagated through the model.
- Part 2. Data preprocessing. This code converts raw data into a DataLoader that can easily do batch processing.
- Part 3. Training epoch. This section includes the forward propagation, computation of losses, backward propagation, and optimisation steps for each training epoch.

For the first part, the code uses PyTorch's torch.nn.Module [15] class to define the structure of the entire neural network, which is consistent with the one designed in III-B and contains an input layer with 12 feature inputs, two LSTM layers, ReLU activation layers, and a fully connected output layer.

The specific flow of the second part is shown in Figure 5. Firstly, the data is read from the csv file, and after encoding the labels according to their numbers, the 12 feature data are created as the temporal data data_x, and the labels are created as the temporal data data_y. These time series data consist of sliding windows, each with a length of 100 discrete data in this practice, which means that 25,100 discrete data gives 25,000 sliding window samples. After that, these temporal data are transformed into a Tensor, and then finally transformed into the DataLoader used for the actual training.

And the third part is the normal training process. Firstly do the forward propagation based on the structure of the neural network and use method criterion() to calculate the loss. After that use the method backward() to compute the gradient for backward propagation based on the loss. And finally use optimiser.step() to update the parameters of the model based on the calculated gradient.

The methods mentioned above are all provided by the classes in Pytorch [15].

IV. Implementation and Result Analysis

In this section, the noise reduction in II-D will first be validated. Although it is visually observable that the noise reduction process results in a reduction of apparent outliers in the data, the most direct validation method is to train with the data before and after noise reduction respectively, and to perform noisy and clean tests on both models.

Then, start the comparative study of multimodal human activity recognition. According to the grouping information in II-B, the grouped data are divided into training sets and testing sets. Afterwards, model training and testing are performed using these data. Based on the results of the tests, investigate the effects of two additional modalities, attitude and gravity,

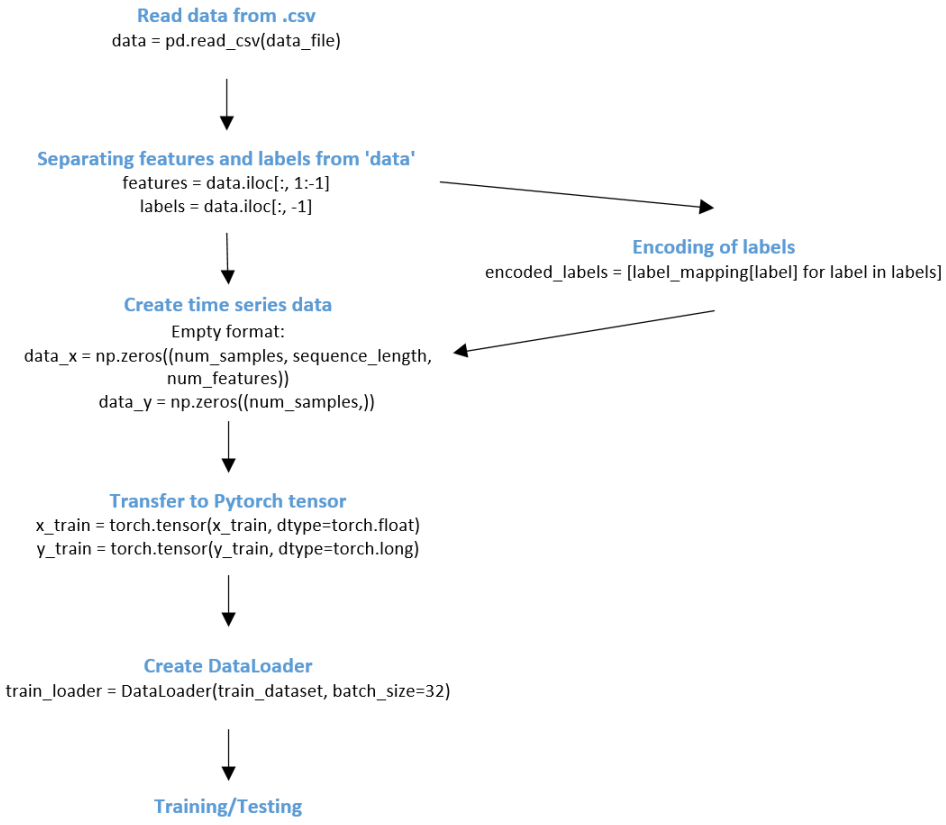


Fig. 5. Data pre-processing (generate DataLoader) flow chart for LSTM training and evaluation. The blue text is the content of the steps, while the black text is some examples of code to show how the data is processed.

on accelerometer- and gyroscope-based multimodal human activity recognition.

During the testing process, in addition to recording the number of successful recognitions, the number and labels of incorrect recognitions are also recorded. From these results, the recall and precision of each human activity for recognition in the corresponding model will be calculated, as shown in Equation (2) and Equation (3).

$$Recall = \frac{TruePositives}{TruePositives + FalseNegatives} \tag{2}$$

$$Precision = \frac{TruePositives}{TruePositives + FalsePositives} \tag{3}$$

The recognition performance of the model in each activity will be considered based on a combination of recall and precision. In this way, to analyse the combined effect of additional modalities on model performance.

A. Noise Reduction Performance Evaluation

As the noise reduction performance was evaluated without considering the effect of modality, both the noisy and clean data used in this case were used in all modalities, with a total of 12 valid variables. After training with noisy and clean data respectively, two models are obtained: the Noisy Training Data Model and the Clean Training Data Model. The two models then do the noisy and clean tests respectively, with a total of 25002 samples, and the recall and precision results are shown in Table II and Table III.

TABLE II
RECALL RATE OF NOISY VERSUS CLEAN TESTS FOR MODELS BEFORE AND AFTER NOISE REDUCTION

Recall Rate	Noisy Training Data Model		Clean Training Data Model	
	Noisy Test	*Clean Test*	*Noisy Test*	*Clean Test*
DWS	93.06%	95.91%	85.07%	96.69%
JOG	93.72%	93.44%	98.06%	97.92%
SIT	100%	100%	99.94%	100%
STD	99.57%	100%	94.08%	100%
UPS	93.82%	93.92%	91.39%	94.95%
WLK	74.29%	73.08%	84.49%	85.09%

Firstly, comparing the noisy test and clean test in Nosiy Training Data Model, it can be found that the noise-reduced data can slightly improve the precision and recall of some activities in the model before noise reduction, such as the activities DWS, SIT, and STD. However, some of the noise-reduced activity data have slightly decreased in precision or recall, such as JOG's recall, UPS's precision, and WLK's

TABLE III
PRECISION OF NOISY VERSUS CLEAN TESTS FOR MODELS BEFORE AND AFTER NOISE REDUCTION

Precision	Noisy Training Data Model		Clean Training Data Model	
	Noisy Test	*Clean Test*	*Noisy Test*	*Clean Test*
DWS	79.40%	79.95%	79.28%	82.88%
JOG	94.99%	96.28%	89.93%	98.31%
SIT	99.99%	100%	100%	99.99%
STD	99.01%	98.98%	99.97%	99.98%
UPS	88.57%	87.66%	93.84%	97.79%
WLK	95.68%	97.69%	91.61%	98.26%

recall.

Then, comparing the noisy tests of the two models, it can be found that although Clean Training Data Model significantly improves the recall of WLK (74.29% to 84.49%), in general, both in terms of precision and recall, Clean Training Data Model's noisy test is not satisfactory, and the overall performance is reduced.

Next, comparing the clean tests of the two models reveals that the Clean Training Data Model has significantly improved performance in terms of precision and recall in identifying all activities, especially the recall of WLK (73.08% to 85.09%).

If the cases of pure noise data and pure clean data are simulated, comparing the noisy test of the Noisy Training Data Model with the clean test of the Clean Training Data Model, it can be found that the overall performance of the case of pure clean data is also better than the performance of pure noise data.

To conclude, there is not much difference between the noisy test and clean test results of Nosiy Training Data Model, which can indicate that the noise reduction does not seriously change the original distribution of the data. Although Clean Training Data Model does not perform well in the noisy test, Clean Training Data Model has a significant performance improvement in the clean test compared to Nosiy Training Data Model.

Based on these findings, the effectiveness of noise reduction can be verified, but if Clean Training Data Model is used, the test data input to the model must also be noise reduced, otherwise the model's comprehensive performance is not as good as Nosiy Training Data Model.

B. Multimodal Recognition Analysis: Confusion Matrix

Based on the grouping, 4 models of different modalities were trained, and the model performance was evaluated using the datasets with a total sample number of 25002 for the different modalities. During the evaluation process, the complete identification results will be recorded, including the number of successful identifications, the number of incorrect identifications, and the misclassification results. Also, the precision and recall will be calculated based on these results as shown in Tables IV, V, VI, and VII. The complete evaluation results allow for a more detailed error analysis of the multimodal human activity recognition process and also can help a comparison investigation of 4 different modalities' model performance.

Firstly, putting all the result tables together and comparing their precision and recall, it can be intuitively noticed that the 4th group, using all the modalities with 12 variables for recognition performs best.

However, recognition performance does not increase linearly based on the addition of modalities. Adding either the GRA or ATT modality to the ACC+GYRO modality in Group 1 makes the recall of WLK activities very bad. For example, in Group 2, after adding GRA modality, the recall of WLK drops from 82.01% to 70.03%; in Group 3, after adding ATT modality, the recall of WLK drops from 82.01% to 79.77%. The results of WLK activity recognition in Group 3 are better than those of Group 2, and are not exaggeratedly different from those of Group 1. Moreover, the precision of WLK in group 2 and group 3 is also slightly reduced.

As the recall rate of WLK activities becomes bad in Groups 2 and 3, the number of WLK activities incorrectly recognised as other activities become more frequent, leading to a decrease in the overall precision in Groups 2 and 3. However, the recall of Groups 2 and 3 is slightly improved compared to Group 1, especially for two activities, SIT and STD.

For some specific activities, the evaluation results vary between each other. Like SIT and STD, due to the specificity of their activities, the data is generally smooth and the recognition performance is very good. Except for the first group, all the other groups have 100% recognition recall, and the precision is also ideal.

WLK is the most difficult to recognise, the best recognition recall is 85.09% in group 4, but the precision of WLK recognition is very high. WLK is easily misclassified as DWS and JOG, and it is misclassified as DWS more often. As the number of modalities increases, the number of times WLK is misclassified as JOG is decreasing, but the number of times WLK is misclassified as DWS is still high. Group 2's misclassification as JOG in recognising WLK became worse after the addition of gravity data to the model.

UPS is also frequently misclassified as DWS, but the overall recognition accuracy and recall for UPS are quite good. Due to the frequent misclassification, while the recall of DWS is ideal, the precision of DWS is the worst of all activities. Overall, the recognition performance of DWS is enhanced with more modalities.

According to the actual association, if UPS, DWS, and WLK with similar speed in the activity are grouped into a new action "Strolling", and taking the data of group 4 as an example for precision and recall calculation, the precision of "Strolling" is 99.31%, and the recall rate is 99.43%, which is very satisfactory.

C. Multimodal Recognition Analysis: SHAP

After a preliminary multimodal analysis using the confusion matrix, it can be seen that WLK is an unexpected result that is more difficult to interpret through the confusion matrix. In a WLK activity based on accelerometer and gyroscope data,

TABLE IV
MULTIMODAL HAR CONFUSION MATRIX FOR GROUP 1

Group 1: ACC+GYRO	***Predicted DWS***	***Predicted JOG***	***Predicted SIT***	***Predicted STD***	***Predicted UPS***	***Predicted WLK***	**Recall**
DWS	23601	268	9	19	813	292	94.40%
JOG	619	24212	0	0	119	52	96.84%
SIT	0	0	24011	991	0	0	96.04%
STD	0	0	433	24569	0	0	98.27%
UPS	1766	30	0	0	23061	145	92.24%
WLK	3034	930	4	41	488	20505	82.01%
Precision	81.33%	95.17%	98.18%	95.90%	94.20%	97.67%	

TABLE V
MULTIMODAL HAR CONFUSION MATRIX FOR GROUP 2

Group 2: ACC+GYRO+GRA	***Predicted DWS***	***Predicted JOG***	***Predicted SIT***	***Predicted STD***	***Predicted UPS***	***Predicted WLK***	**Recall**
DWS	23808	410	24	17	395	348	95.22%
JOG	606	24311	3	0	70	12	97.24%
SIT	0	0	25002	0	0	0	100%
STD	0	0	0	25002	0	0	100%
UPS	1033	218	0	0	23193	558	92.76%
WLK	3696	3650	0	0	148	17508	70.03%
Precision	81.69%	85.04%	99.89%	99.93%	97.43%	95.02%	

TABLE VI
MULTIMODAL HAR CONFUSION MATRIX FOR GROUP 3

Group 3: ACC+GYRO+ATT	***Predicted DWS***	***Predicted JOG***	***Predicted SIT***	***Predicted STD***	***Predicted UPS***	***Predicted WLK***	**Recall**
DWS	23257	513	0	9	865	358	93.02%
JOG	347	24395	10	0	222	28	97.57%
SIT	0	0	25002	0	0	0	100%
STD	0	0	0	25002	0	0	100%
UPS	1182	208	0	0	22763	849	91.04%
WLK	3807	770	0	0	482	19943	79.77%
Precision	81.34%	94.24%	99.96%	99.96%	93.55%	94.17%	

TABLE VII
MULTIMODAL HAR CONFUSION MATRIX FOR GROUP 4

Group 4: ALL	***Predicted DWS***	***Predicted JOG***	***Predicted SIT***	***Predicted STD***	***Predicted UPS***	***Predicted WLK***	**Recall**
DWS	24174	189	3	6	274	356	96.69%
JOG	493	24483	0	0	21	5	97.92%
SIT	0	0	25002	0	0	0	100%
STD	0	0	0	25002	0	0	100%
UPS	1090	157	0	0	23740	15	94.95%
WLK	3411	75	0	0	242	21274	85.09%
Precision	82.88%	98.31%	99.99%	99.98%	97.79%	98.26%	

adding either attitude or gravity data alone will make the recall of WLK low. However, adding both attitude and gravity would make the recall higher. In order to better analyse this situation, two new methods will be introduced:

- 1. Using a larger test set for testing. The confusion matrix in IV-B used only 25002 samples even though some activities with larger test sets. This is to standardise the number of test samples for each action. This time the full WLK test samples will be used, 65,173 in total.
- 2. Introduce the SHAP values of the features for global interpretation.

A table of recall for WLK actions using a larger test set is shown in Figure VIII. Firstly, based on the results in Table VIII, it can be noticed that the most obvious change is that the recall of identifying WLK activities is higher for Group 3 than the results for Group 1, which is a little different from the previous results. In the previous test set with a sample size of 25,002, the recall of WLK for group 3 compared to group 1 decreased from 82.01% to 79.77%. Although the recall rate decreased, the overall difference in recall rate was not significant. However, after using the larger dataset, the recall of all the groups decreased, while the recall of group

TABLE VIII
IMPLEMENTATION RESULTS FOR 65,173 WLK SAMPLES ACROSS ALL GROUPS

Sample Number: 65173	***Predicted DWS***	***Predicted JOG***	***Predicted SIT***	***Predicted STD***	***Predicted UPS***	***Predicted WLK***	**Recall**
Group 1 WLK	12387	1716	4	41	6069	44956	68.98%
Group 2 WLK	11365	10075	0	0	4955	38778	59.50%
Group 3 WLK	7559	2692	34	0	8902	45986	70.56%
Group 4 WLK	7393	2210	0	0	6781	48789	74.86%

1 decreased more and the recall of group 3 decreased less. From the results, 68.98% compared to 70.56% is still similar to before, which is not a big difference, so the models of these two groups have similar performance in WLK activity recognition. For Groups 2 and 4, the results for Group 2 are still the worst and the results for Group 4 are still the best. The gravity data still makes it worse for the model to misclassify WLK as JOG when recognising it.

Next, using SHAP for global interpretation allows for a more detailed analysis of the contribution of individual modalities to recognition. SHAP, standing for SHapley Additive exPlanations, can be viewed as a game theory-based machine learning explanatory framework [16]. SHAP is very versatile and can be used for different scenarios and goals, but for this study, only the global interpretation of SHAP will be used. In the global interpretation of SHAP, SHAP can assign a SHAP value to every single feature, which responds to the magnitude of the feature's influence on the model's predicted result [16]. In practice, it is generally accepted that the larger the SHAP value of a feature, the greater the degree of influence of the feature on the predicted results.

If using SHAP for analysis, visualising the results is a very important step. The ideal visualisation image that facilitates analysis is 3-dimensional and includes the 3 axes of number of samples, number of features, and SHAP value. The number of samples for SHAP is fixed, and in practice, one batch from a DataLoader data is selected as the number of samples, while the batch size of the model is 32, so the number of samples is also 32. In addition to the number of samples, another parameter is the number of background samples. The background samples are the reference dataset used to calculate the SHAP values. In practice, 60000 was chosen as the background sample size, which basically covers the entire test set. For the number of features, the number and order of features are consistent with the input data, with a total of 12 and the order shown in Table I.

The time-series dataset used in this study is two-dimensional with a length of 100 discrete samples and a width of 12 variables of modality. However, SHAP calculates a SHAP value for each feature [16], which results in 1,200 SHAP values in one sample of temporal data and is presented in a two-dimensional matrix. In order to be able to present the results as a 3-dimensional image, the raw SHAP result need to be processed. In practice, adding up the 100 features for each variable in each sample temporal data compresses the data into one-dimensional data of length 12. After processing, a 32*12*SHAP 3D image is obtained, as shown in Figure 6.

Analysing the data in Figure 6, it can be found that the SHAP values of gyroscope and accelerometer are higher compared to gravity and attitude, representing that gyroscope and accelerometer data are more important in the recognition process. Among them, gyroscope contributes more than accelerometer.

If comparing the SHAP means of Group 1 and Group 2, it can be seen that Group 2, with the addition of the gravity modality, reduced the SHAP means of the gyroscopes and accelerometers, which are key to the recognition, to a lower value. The gravity modality affects the contribution of gyroscopes and accelerometers to the recognition of WLK activities to some extent, and the SHAP mean of gravity is very low and does not contribute much to the recognition. This is the main reason for the low recognition recall in Group 2.

Comparing the SHAP mean values of Group 1 and Group 3, the SHAP mean values of gyroscope and accelerometer also decreased in Group 3, but the SHAP value of the most contributing gyroscope data was higher than that of Group 2. Moreover, the SHAP value of attitude is greater compared to gravity. With the combined effect, the performance of Group 3 is similar to Group 1 with similar recall.

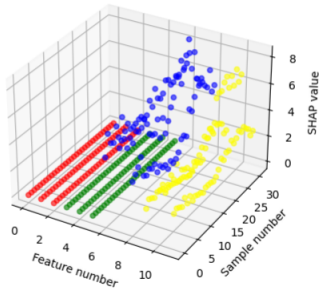
The best-performing group 4, on the other hand, has a higher SHAP mean than groups 2 and 3, although the SHAP means of the gyroscopes and accelerometers are still down compared to group 1, and the SHAP means of some individual gyroscope feature is even higher than those of group 1. There are also more effective attitude data contributing to the recognition. Taken together, Group 4 has the most favourable recall.

V. CONCLUSION

In conclusion, according to the results of the confusion matrix, the performance of the model does not increase linearly according to the increasing number of modalities. Adding gravity alone to gyroscopes and accelerometers makes the model lower performance and the worst performing model. Whereas the model with attitude data added has about the same performance as the model with only gyroscopes and accelerometers. The model performs best when all modalities are used, and even after combining WLK, UPS, and DWS into a new action called "Strolling", the precision and recall of each action exceeds 97.5% in a test set of 25002 samples.

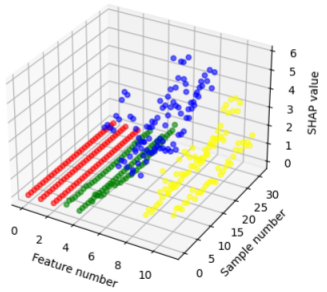
After using a larger dataset and introducing SHAP values for global interpretation of WLK activity, it can be observed that gyroscopes and accelerometers contribute the most to

(a)



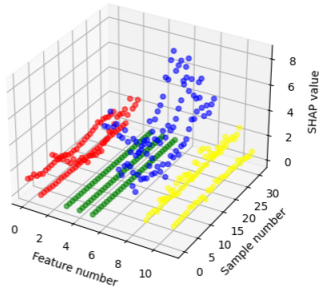
Feature:	1	2	3	4	5	6	7	8	9	10	11	12
Maximum	0	0	0	0	0	0	8.17	8.19	6.85	3.46	7.39	3.61
Mean	0	0	0	0	0	0	5.76	5.84	3.97	2.19	4.51	2.39
Mean	0	0	0	0	0	0	4.44	3.68	2.01	1.27	2.51	1.50

(b)



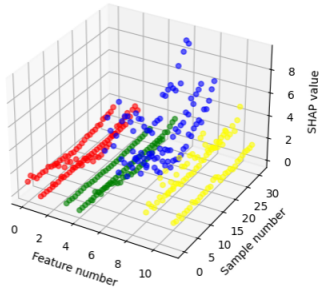
Feature:	1	2	3	4	5	6	7	8	9	10	11	12
Maximum	0	0	0	0.23	0.06	1.02	5.82	4.67	4.77	1.43	3.81	1.90
Mean	0	0	0	0.13	0.04	0.57	4.03	3.06	3.36	1.13	2.73	1.38
Mean	0	0	0	0.07	0.03	0.30	2.34	2.16	2.08	0.82	1.72	0.96

(c)



Feature:	1	2	3	4	5	6	7	8	9	10	11	12
Maximum	1.22	0.26	3.71	0	0	0	8.43	6.66	5.37	1.07	3.62	1.43
Mean	0.74	0.18	1.98	0	0	0	5.53	4.30	3.24	0.82	2.67	1.09
Mean	0.34	0.10	0.80	0	0	0	2.91	2.70	1.55	0.59	1.95	0.69

(d)



Feature:	1	2	3	4	5	6	7	8	9	10	11	12
Maximum	1.42	0.20	3.14	0.42	0.05	1.96	9.38	5.73	6.28	2.19	5.34	1.90
Mean	0.78	0.11	1.63	0.22	0.03	1.04	6.35	3.87	3.78	1.61	3.85	1.47
Mean	0.33	0.08	0.55	0.10	0.02	0.40	3.85	2.26	2.13	1.22	2.73	1.09

Fig. 6. Visual representation of SHAP values for (a) Group 1 (b) Group 2 (c) Group 3 (d) Group 4 in all modalities. Features 1 to 3 are ATT (Attitude), shown in red on the graph; features 4 to 6 are GRA (Gravity), shown in green on the graph; features 7 to 9 are GYRO (Gyroscope), shown in blue on the graph; and features 10 to 12 are ACC (Accelerometer), shown in yellow on the graph.

the evaluation results, with gyroscopes contributing more than accelerometers. Gravity contributes the least to the evaluation results and also decreases the contribution of gyroscopes and accelerometers. Attitude also reduces the contribution of the gyroscopes and accelerometers, but attitude contributes more to the results than gravity. The contribution of the gyroscopes and accelerometers is also reduced when using all modalities, but the degree of the reduction is significantly smaller and, together with the contribution of the attitude, gives the best recognition performance for the model using all modalities.

The gravity data identifies a small contribution of WLK activity and may lead to a worse misclassification of WLK as JOG. My conjecture is that since the data was measured from a mobile phone placed in a trouser pocket, even though the leg lifting amplitude is different between walking and jogging, the frequency of jogging is much faster, so there is a repetitive and highly similar portion of the timing data and walking. This conjecture needs to be verified by further empirical studies.

REFERENCES

[1] Baltrušaitis, T., Ahuja, C., & Morency, L. P. (2018). Multimodal machine learning: A survey and taxonomy. IEEE transactions on pattern analysis and machine intelligence, 41(2), 423-443.

[2] Ramachandram, D., & Taylor, G. W. (2017). Deep multimodal learning: A survey on recent advances and trends. IEEE signal processing magazine, 34(6), 96-108.

[3] Zhang, S., Li, Y., Zhang, S., Shahabi, F., Xia, S., Deng, Y., & Alshurafa, N. (2022). Deep learning in human activity recognition with wearable sensors: A review on advances. Sensors, 22(4), 1476.

[4] Ferrari, A., Micucci, D., Mobilio, M., & Napoletano, P. (2019, November). Human activities recognition using accelerometer and gyroscope. In Ambient Intelligence: 15th European Conference, AmI 2019, Rome, Italy, November 13–15, 2019, Proceedings (pp. 357-362). Cham: Springer International Publishing.

[5] Malekzadeh, M., Clegg, R. G., Cavallaro, A., & Haddadi, H. (2019, April). Mobile sensor data anonymization. In Proceedings of the international conference on internet of things design and implementation (pp. 49-58).

[6] Blázquez-García, A., Conde, A., Mori, U., & Lozano, J. A. (2021). A review on outlier/anomaly detection in time series data. ACM Computing Surveys (CSUR), 54(3), 1-33.

[7] Apple. (2023). CMDeviceMotion. [Online]. Developer. Last Updated: 2023. Available at: https://developer.apple.com/documentation/coremotion/cmdevicemotion [Accessed 1 August 2023].

[8] De, M. (2021). Tukey Fences for Outliers. [Online]. IBM. Available at: https://community.ibm.com/community/user/ai-datascience/blogs/moloy-de1/2021/03/23/points-to-ponder [Accessed 2 August 2023].

[9] Yu, Y., Si, X., Hu, C., & Zhang, J. (2019). A review of recurrent neural networks: LSTM cells and network architectures. Neural computation, 31(7), 1235-1270.

[10] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.

[11] Dhruv, P., & Naskar, S. (2020). Image classification using convolutional neural network (CNN) and recurrent neural network (RNN): A review. Machine Learning and Information Processing: Proceedings of ICMLIP 2019, 367-381.

[12] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

[13] Yin, W., Kann, K., Yu, M., & Schütze, H. (2017). Comparative study of CNN and RNN for natural language processing. arXiv preprint arXiv:1702.01923.

[14] Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In Proceedings of the 27th international conference on machine learning (ICML-10) (pp. 807-814).

[15] PyTorch. (2023). PyTorch documentation. [Online]. PyTorch. Available at: https://pytorch.org/docs/stable/index.html [Accessed 9 August 2023].

[16] Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. Advances in neural information processing systems, 30.