STUDY AND APPLICATION OF AI TECHNIQUES FOR ACCURATE HUMAN ACTIVITY RECOGNITION

A DISSERTATION

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF DEGREE
OF

MASTER OF TECHNOLOGY

IN

SIGNAL PROCESSING AND DIGITAL DESIGN

Submitted By:

LOKESH DHAMMI
(ROLL NO. 2K19/SPD/08)

Under the supervision of

MR. PIYUSH TEWARI

(ASSISTANT PROFESSOR, ECE DEPT)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

JULY 2021

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CANDIDATE'S DECLARATION

I **LOKESH DHAMMI**, Roll No. 2K19/SPD/08 student of M.Tech (Signal Processing & Digital Design), hereby declare that the project Dissertation titled **"STUDY AND APPLICATION OF AI TECHNIQUES FOR ACCURATE HUMAN ACTIVITY RECOGNITION"** which is submitted by me to the Department of Electronics and Communication Engineering, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Dhammi

Place: Delhi **NAME: LOKESH DHAMMI**

Date: 28/07/2021 IVth- Sem, M. Tech (SPDD)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CERTIFICATE

I hereby certify that the Project Dissertation titled **"STUDY AND APPLICATION OF AI TECHNIQUES FOR ACCURATE HUMAN ACTIVITY RECOGNITION"** which is submitted by **LOKESH DHAMMI**, **2K19/SPD/08** of Electronics and Communication Department, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology, is a record of the project work carried out by the student under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Place: Delhi

Date: 28/07/2021

Mr. PIYUSH TEWARI

ASSISTANT PROFESSOR,
ECE DEPT.

ABSTRACT

HAR is the fastest-growing field of Computer Vision (CV) that has a broad range of applications. Recognition of human activities is a very challenging task as we are moving towards automation and AI. In this thesis, we have proposed an intelligent smartphone-based human activity recognition system using deep learning techniques. HAR system can automatically classify and predict the daily human activities gathered using the inbuilt smartphone sensors like gyroscope and accelerometer. The proposed deep learning models are the CNN and the LSTM network. The proposed models are evaluated on WISDM dataset. Moreover, Batch-Normalization (BN) and Dropout layers are used in between the proposed network to reduce the overfitting of models.

The proposed models have shown good results with high accuracy and low complexity. 'CNN' model achieves accuracy up to 95% and the 'LSTM' model achieves accuracy up to 97.5%. The complexity of the proposed network is low with 21,922 as trainable parameters. Experimental results are depicted with 'Learning curves' and 'Confusion matrix'. The results of this research are quite promising and has no limitation to utilize on environment.

ACKNOWLEDGEMENT

I thank the almighty for giving us the courage and perseverance in completing the Dissertation. This project itself is acknowledgements for all those people who have given us their heartfelt co-operation in making this project a grand success.

I would like to thank Head of the E.C.E. Department, **Prof. N.S. RAGHAVA** for providing constant encouragement. sufficient infrastructure and good environment in the college to complete our course.

I am greatly indebted to project guide **Mr. PIYUSH TEWARI**, Assistant Professor, Electronics & Communications Engineering, for providing valuable guidance at every stage of this project work. I am profoundly grateful towards the unmatched services rendered by him.

Our special thanks to all the faculty of Electronics & Communications Engineering and peers for their valuable advises at every stage of this work.

Most importantly, none of this would have happened without the love and patience of family. I would like to express my deep sense of gratitude and earnest thanksgiving to **My Dear Parents** for their moral support and heartfelt cooperation in doing the project.

Dhammi

LOKESH DHAMMI

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1. HUMAN ACTIVITY RECOGNITION (HAR)

The ‘Human Activity Recognition (HAR)’ system aims to classify and recognize the human activities gathered with the help of sensors. In the generation of IoT and AI, detecting human actions is in demand for researchers and industries.[1]

Activities of individuals are divided into four different parts:

Gesture: it the movement with the body part which gives meaningful expression and it can be seen from the naked eyes. For example – sign expressions for disabled, showing like and dislike.

Action: it is shown by only one person which expresses some activity. For example-running, going upstairs.

Interactions: Activity done by two people or interacting with the object, performing some action together. For example- talking with each other, shaking hands.

Group activities: Activities performed by the group of people showing some action performing altogether. For example- playing hockey, playing polo.

HAR system is a very leading area of research and used in many different use cases. HAR system is used for a variety of applications like monitoring the elder patients, human-computer interaction, smart living, video surveillance, gaming, monitoring the daily performance of sports-person, entertainment, etc [2][3]. HAR system is continuously making our daily life much smarter and safer which also improves the quality of service towards society.

In the field of HAR and CV, a lot of results had been published and verified, implementing different technologies which results in continuous improvement in performance of the system. Modern technologies like RNN, DL, CNN that are extracting and learning the features in very depth are changing the world of AI, CV and HAR. Research in the field of classifying the activity is increasing because due to this a lot of accidents had been evaded. However, modeling activity recognition and generating to the point and efficient output is still a big task due to variations in different parameters like silhouette, deformation, illumination, different angles, etc.

Based on data collected for detection of activities, HAR can be divided into two parts:

- **a. Video-based**
- **b. Sensor-based**

Videobased HAR system includes the input data (raw data) which is collected in the form of videos. Videos are pre-processed and converted into short clips which represent different human activities. With the help of videos, human activities are recognized and detected. These videos can be taken from different types of cameras, used according to the end application.

In a Sensor-based HAR system, smart sensors are used to collect the raw data which some or other way represents the activities of a human being. Usually, sensors gather the data and represent it in numeric form or the form of signals (waveforms). The particular pattern of the signal represents one activity performed by the subject. Sensors can be of many types like sound sensors, gyroscope, microphone, Bluetooth, accelerometer, piezoelectric sensor, etc.

Sensor-based HAR system has many advantages over Video-based HAR system. Research in the field of 'sensor-based' technology, is increasing rapidly as sensors are a basic and important part of every electronic-based project. Advancement in circuit optimization technologies has led the sensors to fit in every smart device which is capable to do multiple tasks.

When we talk in terms of execution time, a video-based HAR system takes a long time to execute a full project while a sensor-based HAR system takes a shorter time. Latest modern developments in sensor-based technologies have solved the problem of anti-magnetic field interference [4]. The problem of antimagnetic field interference occurs when sensors like accelerometer and gyroscope are unable to sense the accurate values of acceleration(accelerometer) and angular velocity(gyroscope) because of magnetic field present which cause interference.

Sensors can be of two types based upon their installation:

i. **Fixed Sensor** – Fixed sensors are non-movable and installed in only one place. They need special equipment for installation. For example- acoustic sensors [5][6], vibrational [7], static cameras [8], etc.

ii. **Motion Sensors** – Motion sensors are movable sensors; they are not bounded at one place and their sensing activity is not confined to a limited area.

The basic flow of the HAR system is represented in Fig.1. The following steps explain the flow of the HAR system:

a. Data collection:

The foremost and the primary step in the implementation of the HAR system is to collect the relevant information for recognition of human activities. This can be done with the help of various types of sensors available in the market. Sensors can be fixed or motion sensors and can be body-worn or sensors in smart devices. Information received by sensors is in the form of images or videos or numerical data. This data is raw input data and can be used for different end applications.

b. Data Pre-processing:

The raw-data gathered in the above step is usually in JSON format. JSON stands for Java Scripted Object Notion. File in JSON format is very unstructured and unbalanced which contains a lot of unnecessary information. So, we need to balance it by extracting meaningful information from it. Also, we convert the information and represent it as waveforms (signals). These waveforms represent the unique pattern for a particular activity. Fig. 2 represents the waveforms for 6 different human activities using the accelerometer sensor.

c. Split into test and train set:

After we get the clean and structured dataset we will split it into two parts ‘train-set’ and ‘test-set’. Usually, we divide the dataset into a 70:30 ratio i.e., 70% of ‘train-set’ and 30% of ‘test-set’. We will train our model on the train data (70% of the total dataset) so that model will learn all the patterns and then test it on the test-set (30% of the total dataset).

d. **Feature Extraction and Classification:**

Different ‘feature-extraction’ techniques are used to bring out the important ‘deep features’ from the dataset which is helpful in the accurate detection of human activities. Also, various ‘classification-techniques’ are used to classify human activities. For example- ANN, CNN, Deep Learning, Machine Learning, HoG, etc. Our model learns the features with the help of a training set and then human activities are classified using a test set.

Activities can be classified into three types on the basis of their duration of transition time and complexity.

i. **Short Activities:**

Short activities have a very short duration of transition time for example- sitting to laying, stand to walk, etc.

ii. **Simple Activities:**

Simple activities have a long duration of transition time. They are easy to understand and easily recognizable. Simple activities include- running, reading, writing, etc.

iii. **Complex Activities:**

Complex activities are those which include the interaction of two or more subjects (group activity). They are difficult to recognize. E.g. group activities including two or more human beings.

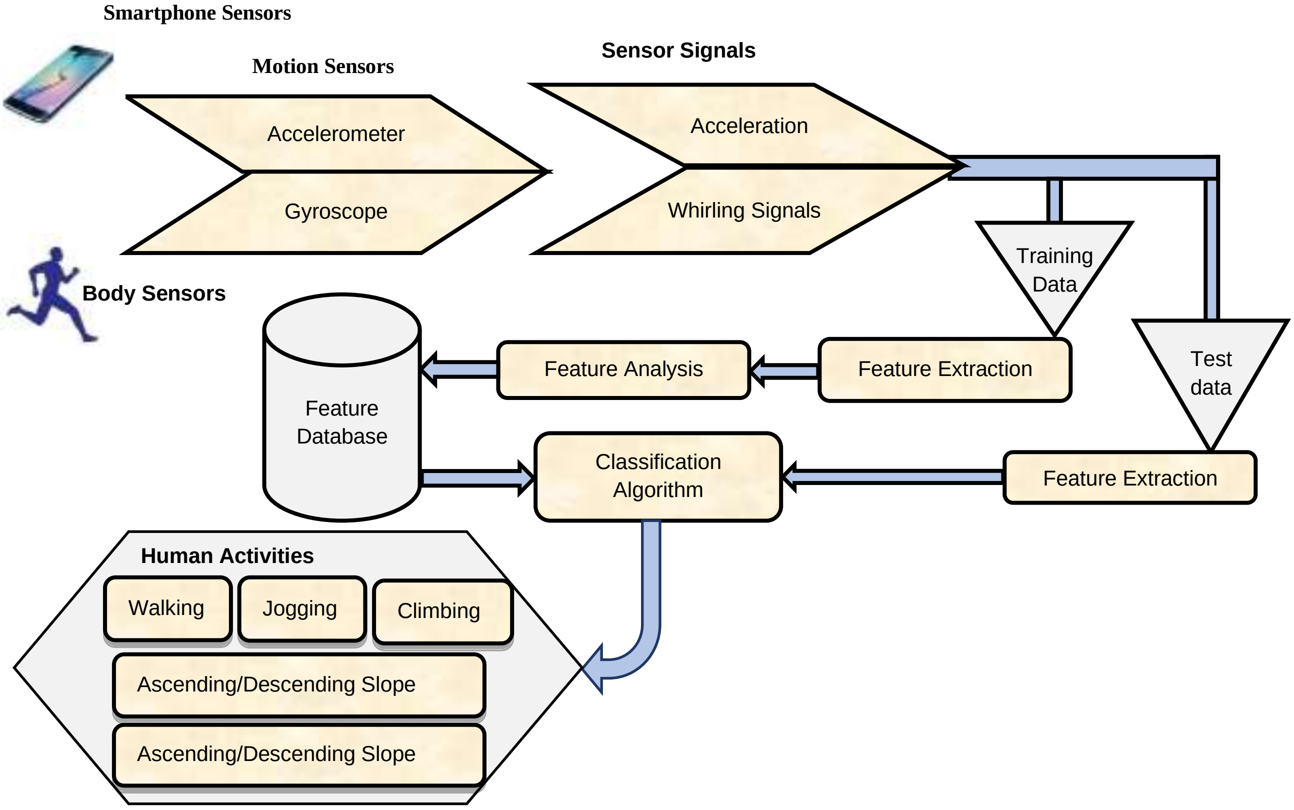


Fig. 1.1 Flow of HAR system.

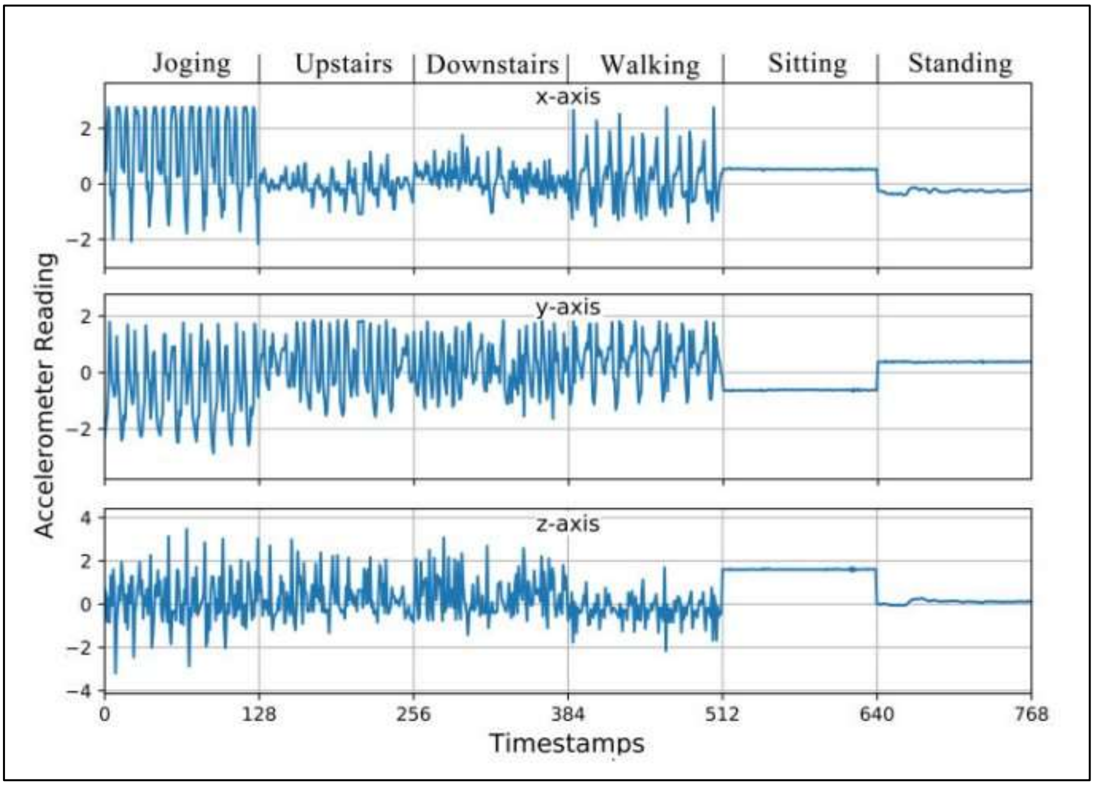


Fig. 1.2 Activities represented in waveforms.

CHAPTER 2

BACKGROUND

2.1 HAR RESEARCH AREAS

HAR is used in many different industrial applications and healthcare applications.

2.1.1 CONSUMER ELECTRONICS

Day-to-day lifestyle activities and the routine of elder patients or human beings show their interconnection with the gadgets which are all around them, it can be smart lights, smartphones, laptops, iPad, etc.

These interconnections of devices with each other and human being is called as the Internet of Things as they are interlinked with each other with the help of internet. These devices are consumer electronic devices.

Overuse of these can be harmful to health. The study says that prolonged use of laptops and mobiles can deteriorate eyesight. Also using electronic devices in the wrong posture can cause stress, neck pain, obesity and many other forced posture variations in the body.

Therefore, monitoring the activities can reduce stress and other negative effects of prolonged usage of devices.

Recognizing all the anomalous activities such as playing polo, playing golf, bending backward, bending forward, climbing, running & standing from bed in all these activities kinetic camera is used. A kinetic camera is used to click the depth images of the activities. To perceive the activities captured from the cameras some feature extraction techniques are used like R-transform, Convolutional Neural network.

2.1.2 HEALTH CARE MONITORING

The activity recognition field is the foundation stone for smart health care monitoring which uses the IoT, AI, ML, DL and RNN in its applications.

Monitoring and predicting the activities of patients and other human beings have increased their ‘living standard’ and ‘quality of life’.

For the patients', activities like taking medicines on time, meals on time, not to exaggerate much, not to be a movement for a long time for the heart patients.

Object detection or fall detection systems can sensor the activity. Some techniques used in this area of research are RFID, NFC, short-range wireless transmission, etc.

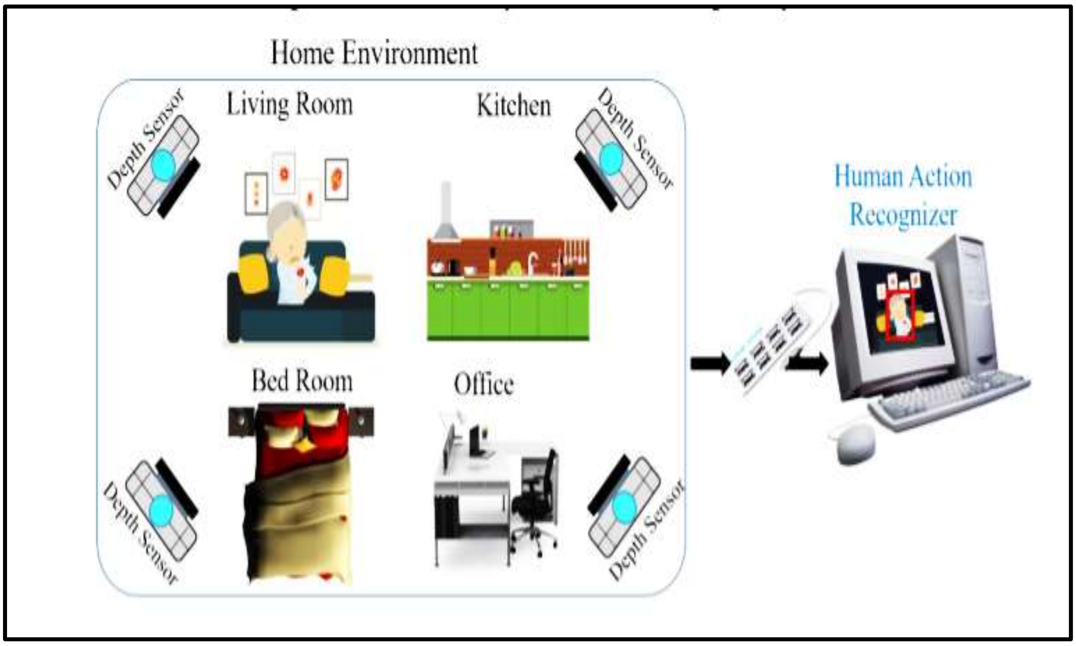


Fig 2.1 Monitoring the activities [1]

2.1.3 SECURITY AND SURVEILLANCE

Activity recognition performs the major task in the field of security, many defense purposes, traffic controlling, spying, and many more.

For example, at public places like malls, railway stations, bus stations and religious places(temples) has many visitors and there is a possibility of loitering people, theft, findings of bomb or other suspicious activity and objects. This can be detected using this technique of action detection and object detection.

A smart detection system has been developed which can automatically detect the uncommon activity in that particular area with the help of real time video capturing.

2.1.4 HAR WITH DEEP LEARNING

Action recognition and trending deep learning research areas go simultaneously with each other. There are many different types of deep learning networks and sub-techniques which is applied for activity classification application. SVM-support vector machine, Deep CNN, RNN, residual networks (skip connections) are some of the examples of techniques

Active participation of senior citizens in society is very important from the social and economic point of view. Currently monitoring the daily activities of an elder person increases their safety and well-being. And this area of computer vision is very interesting and challenging which need to be explored more (especially for wearable sensors)

2.2 LITERATURE REVIEW

Detecting human activities is a very challenging and important task for researchers. Researchers are getting very frequent positive outcomes which takes them close to their goal. Many different challenges and hurdles are present which is to be rectified. Nowadays, the latest technology of hardware and software are present which will ease the evaluation of models especially if we compare with earlier technology in the 90's.

During earlier research in the field of human activity detection, researchers recognize the simple and short activities. Short activities include those activities which have very short transition time e.g., handshaking, sitting to laying, etc. As research goes on, they started to recognize real-life complex activities like playing, reading, running, etc. Also, they tried to rectify the challenges used in the HAR system. To recognize the simple activities, traditional Machine learning algorithms like Decision trees, KNN, SVM, are used.

L. Bao et al. [9] used 'biaxial' accelerometer sensors to wear on various body parts to recognize the activities. They gathered the data using 20 subjects. The 'Decision-Tree' classifier gives the best results i.e., accuracy of 84%. Also, the result shows that using multiple sensors at different body parts improved the accuracy as it is easy to discriminate between activities.

A. Jain et. al. [10] recognized human activities using a descriptor-based approach. They have used histogram gradient and Fourier descriptor to extract the features and used SVM and k-NN to evaluate the results. P. Gupta and T. Dallas [11] recognized six human activities with triaxial-accelerometer, worn at the waist of the subject. They have used Relief-F and SFFS to extract the features and classify activities using Naïve Bayes and KNN algorithms.

E. Fullerton et al. [12] used machine learning algorithms to detect different human activities in free-living. Ten subjects have worn nine multiple body-worn sensors. Different pre-processing, feature extraction and classification techniques are evaluated, KNN and Decision tree algorithms have shown good results in recognizing the activities.

J. Margarito et. al. [13] have used the template matching approach to detect eight sports-activities using wearable sensors. The triaxial accelerometer is used located at the wrist of subjects, 29 normal weight and 19 overweight. Different statistical learning classifiers like 'Decision Trees' (DT), 'Naïve Bayes', 'Logistic Regression' (LR), and 'Artificial Neural Network' (ANN) are evaluated.

Advanced technologies in Machine Learning, AI, Neural-Networks and Deep-Learning (DL) have shown very good results when applied for HAR tasks. Also, these technologies are able to solve the HAR challenges and able to classify or recognise the practical complex human activities which are used in different real-time applications. Deep Learning technology has attracted many fields like computer vision, image processing, video processing, NLP, etc.

J. Mantyjarvi et al. [14] generated features using 'Principal-Component-Analysis' (PCA) and 'Independent-Component-Analysis' (ICA) along with wavelet transform. Activities are classified using the multilayer perceptron classifier. Y. Zhen et. al. [15] explored and compared different feature learning techniques with traditional feature-based approaches. The authors proposed Deep CNN for the time series classification of human activities. Y. Chen et. al. [16] propose a CNN model to classify eight different human activities with the help of triaxial accelerometer signals. Model is evaluated on the dataset with 31688 samples taken from 100 subjects.

C.A. Ronao et. al. [17] presented an effective and efficient recognition system based on the sensors embedded in smartphones. Authors proposed Deep CNN model which can extract the roburst-features automatically from the input data and got very promising results.

F. Ordonez et. al. [18] used RNN for the HAR task, they applied DeepConvLSTM i.e. a fusion of 'Deep-Convolutional-Neural-Network' and 'Long-Short-Term-Memory' (LSTM) on time series input signal. They evaluated the model on two different datasets and presented accuracy and F1 Score as model performance.

Deep Learning methods are very effective in extracting the deep-features from the raw-input data taken from built-in sensors. Also, deep learning methods had automated the feature extraction process. A. Murad et. al. [19] proposed a deep RNN network model which can recognize and classify human activities. Authors analyzed that it can solve the problem of long-term dependency faced during variable-length input sequence. Authors presented many RNN networks, DRNN outperforms and shown promising results.

Y. Guan et. al. [20] resolved the real-life challenges faced during the implementation of HAR applications. The authors proposed an Ensemble LSTM network that modifies the training pattern for LSTM. They evaluated the model performance on three different datasets- 'PAMAP2', 'OPPORTUNITY' and 'Skoda'. Ensemble LSTM got promising results when compared to individual LSTM networks and other state-of-the-art recognition techniques.

In [21], Lokesh et al. have proposed the smartphone-based intelligent human activity recognition system. They have used a public dataset (WISDM) collected using smartphone sensors (accelerometer and gyroscope). Dataset is converted in the form of signals after pre-processing and apply proposed models. Authors have proposed an LSTM model to classify six human activities. The experimental result shows that the overall complexity of the system is reduced with increased accuracy.

2.3 PROBLEM STATEMENT:

The techniques discussed in previous section has obtained good results but they are very complex in structure. This complexity in the structure reduces the training speed of model. The proposed model has low complexity. The process of collecting the data involves the wearable sensors which are not able to gather the correct values and also, the body worn sensors are uncomfortable for subject to wear on different body locations. Therefore, smartphone sensors are used for data collection as they are portable and handy. The major contributions of this thesis are as followed:

1. The proposed model (LSTM) has high accuracy and recognise the human activities accurately.
2. The complexity of the network is reduced.
3. Less number of parameters which results in low cost and high speed.
4. The optimal use of smartphone sensor- wearable sensors are not good choice for monitoring of activities therefore smartphone sensors are used to collect the activity dataset.

TABLE 2.1. HAR SYSTEM BASED ON CLASSIFICATION TECHNIQUES

Dataset	Classification Technique	Accuracy	Reference
UCI HAR	SVM	97.12 %	[10], [22], [23], [24]
	KNN	91.75 %	
	CNN	92.37%	
	DeepConvLSTM	92.44%	
	LSTM+CNN+GAP	**95.78%**	
	DBN	95.85%	
	MLP	91.7%	

WISDM	KNN	96.2%	[21], [23], [25], [26], [31]
	CNN	93.27%	
	DeepConvLSTM	93.01%	
	LSTM+CNN+GAP	95.85%	
	EnsemConvNet	97%	
	LSTM	**97.5%**	
	CNN	85.1%	
OPPORTUNITY	DeepConvLSTM	91.5%	[23]
	LSTM+CNN+GAP	92.63%	
	KNN	82.86%	
UniMiB SHAR(AF-17)	EnsemConvNet	92.6%	[27], [31],
	Random forest	88.41%	
UniMiB SHAR(A-9)	MBOSS	96.38%	[26], [27], [31]
	EnsemConvNet	**98.7%**	
	SVM	90.55%	
MobiAct	ANN	96.07%	[28], [29], [30], [31]
	CNN+LSTM	**96.9%**	
	EnsemConvNet	95.4%	

CHAPTER 3

METHODOLOGY

3.1 THEORY:

In this chapter, we will discuss all major methodologies and techniques which are used to recognise the human activities. As we have discussed in the previous chapter that initially traditional ML algorithms are used for the recognition of simple and short human activities. So, we will discuss the traditional ML algorithms and modern deep learning algorithms. For the process of data collection, various sensors are used to sense the daily activities of human beings. Usually, accelerometers and gyroscope sensors are used.

3.1.1 SENSING ACTIVITY:

Numerous sensors are generally utilized in HAR and measure different features and important signs for example-temperature, body-blood-pressure, body heart-rate, motion (acceleration, speed) & environmental signal (light intensity and environment) but to decide on the right sensor we'd like to consider the first element for the design of HAR system. The sensor mechanism is classed on the idea of sensor placement concerning the user. If the sensor is found in the environment, then it's ambient and if it is attached on the subjects body, then it's wearable. But as nowadays technology is very advanced, sensors built inside smartphones plays a vital role in sensing activity for the application of HAR systems. In Fig 3.1 we can see that wearable sensors are applied at different body parts like on wrist, on arm, on thighs, etc. Here we will discuss some sensors usually used for sensing human activities. These can be used as body-worn sensors and also, they are embedded into smartphones.

1. Accelerometer sensor
2. Gyroscope sensor

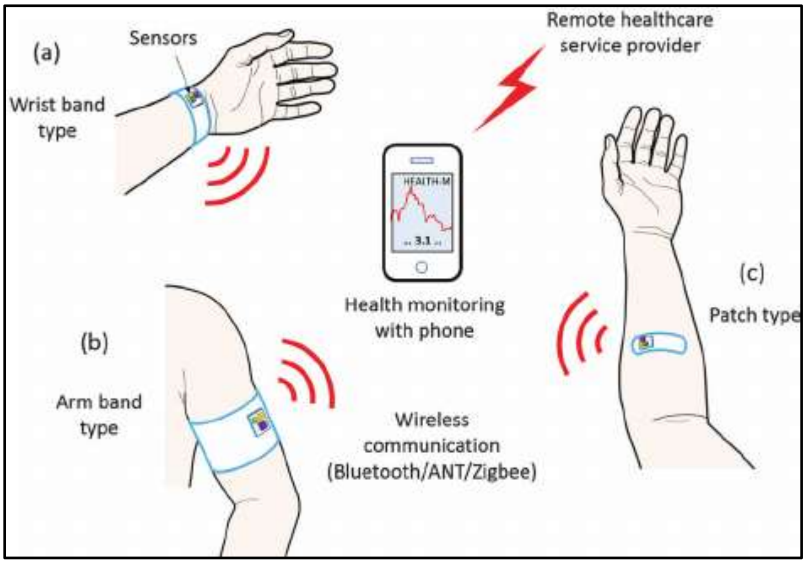


Fig.3.1 Sensors applied on different parts of the body (wearable-sensors)

1. ACCELEROMETER-SENSOR:

The commonly used sensor in the hardware industry is accelerometer sensor. This device measures the physical acceleration of an object. Also, it is used to measure the vibration-in motors, high-speed vehicle and high-loaded bridges, etc. The working of the accelerometer-sensor is based on seismic-mass which gets displaced with reference to accelerated object. The distance measured is converted into the electrical signal. This principle is also applied to MEMS-sensors. This sensor senses all the three-dimensional axis that is x, y and z-axis of the object. It gives numerical data of x, y, z-axis with reference to equator.[32]

2. GYROSCOPE SENSORS:

It can sense the angular velocity and orientation of a subject. due to this reason gyroscope sensor is also known as an angular velocity sensor. In the gyroscope sensor, the angular rate is measured and for that rotation rate is converted into electrical signals.[32]

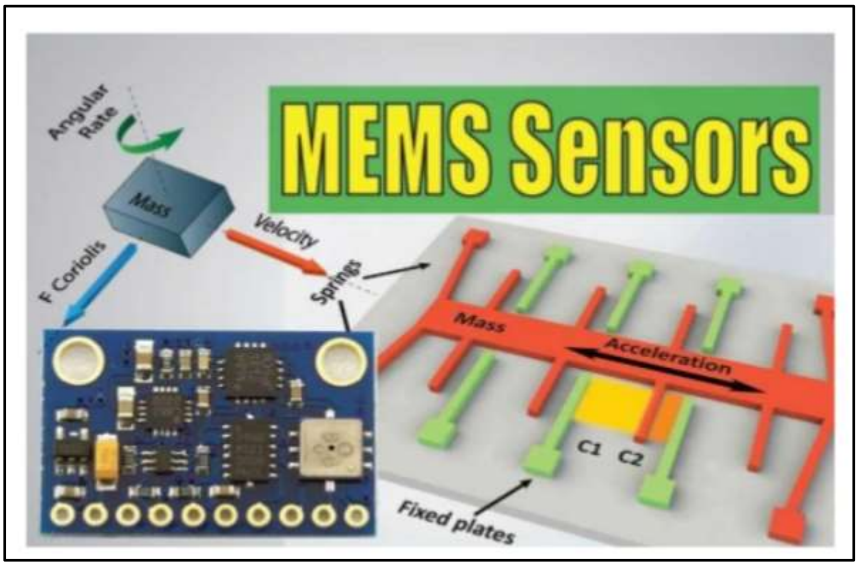


Fig.3.2 Accelerometer sensor

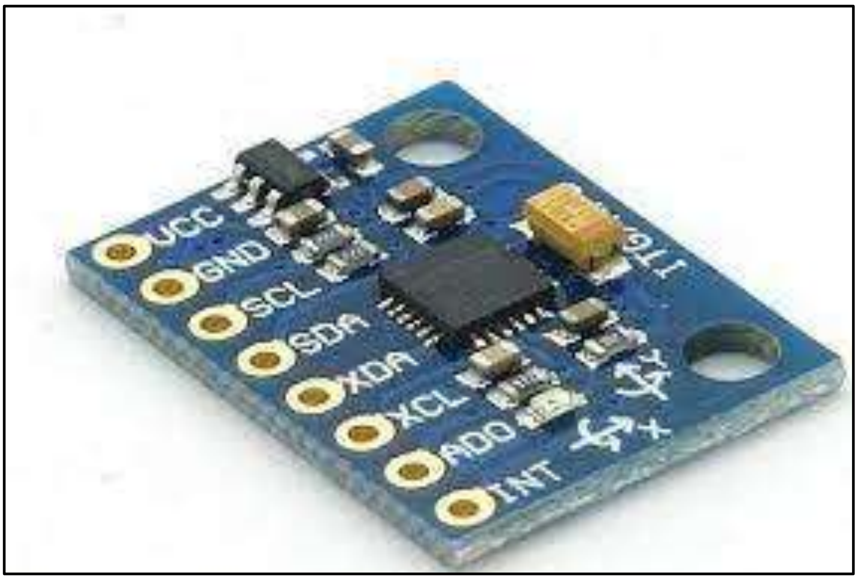


Fig 3.3 Gyroscope sensor module.

3.1.2 MACHINE LEARNING ALGORITHMS:

1. SUPPORT VECTOR MACHINE (SVM):

‘SVM’ algorithm comes under supervised-learning. SVM solves the both regression and classification problem. In the SVM classifier, we construct an ‘N-dimensional’ space where ‘N’ is the no.of features in a particular dataset. We represent each data point in N-D space. Finally, we find the hyper-plane which separates the space into two classes (for two-dimensional space). If the location of the new point is right to the hyper-plane it belongs to 'Class 2' and if the location of the new point is left to the hyper-plane it belongs to 'Class 1'. [33]

2. **DECISION TREE ALGORITHM:**

The Decision-Tree(DL) Algorithm is the type of ML algorithm that handles regression and classification problems. As the name suggests, the Decision Tree Algorithm is represented as an inverted tree i.e., when the tree is represented upside down. The first node is defined as root node and other nodes are connected with the root node with the help of branches. Branches define some conditions to get the next node. The last node or the lower-level nodes are known as leaf nodes or decision nodes as they give the final decision.

To decide the root node, position of other nodes, and the structure of the tree we calculate the Entropy, Information Gain, and Gini impurity. Entropy is the measure of randomness, also, it measures the purity of spilt i.e., how much it is good to divide that particular node. Information Gain helps to decide the type of decision tree structure. A structure having the highest Information Gain is selected. Gini impurity calculates the impurity of a decision tree. [34]

3. **KNN ALGORITHM:**

K-NN is the supervised ML algorithm and handles regression and classification challenges. Whenever we try to classify a new point, we first have to decide the number of nearest-neighbour points. The word 'K' in KNN is a number of nearest neighbor points. We then calculate the distance between the new point and all the nearest neighbours. The distance to be calculated can be Euclidian distance or Manhattan distance. A category which is having the maximum number of nearest neighbours, a new point belongs to that category. [35]

3.1.3 DEEP LEARNING:

In a rapidly growing world of AI, Deep learning techniques are used in every domain whether it is healthcare, entertainment, traffic control, banking, etc. Deep learning has attracted various area of research like in image processing, Natural Language Processing, Video Processing, Computer Vision, etc. Deep Learning tries to work just like a human brain. The human brain learns the features of a subject and feeds that in memory and tries to memorize them. Similarly, Deep Learning techniques automatically extract the meaningful features of input, and based on these features, it classifies the new subject. Whereas, If, we compare machine learning techniques with DL techniques, machine learning techniques are based upon the 'manual-feature-extraction' of input which is of limited knowledge specially when we talk about HAR.

i) CONVOLUTIONAL NEURAL NETWORK (CNN)

CNN is a kind of 'Artificial-Neural-Network' which performs feature extraction and classification of input. CNN has three parts(layers) i.e., convolution, pooling and output-layer. These layers are the fundamental building blocks of CNN which can be repeated multiple times according to the end application.

Convolution

Convolution is a technique that is used for the feature extraction of the input. This is done with the help of kernels that can be of different sizes. Kernels are the matrix that defines the particular feature of input. The function of convolution is to convolve the input with kernels and give the result matrix. The resultant matrix is known as a feature map.

Kernels belong to the particular characteristic which we need to find out from the input. Following Fig 3.4 is an example of convolution in which the input image matrix of 5x5 convolves with the kernel matrix of 3x3 and giving a feature map as an output of matrix 3x3. Fig 3.5 represents the example of edge detection in the RGB input image.

Convolution can be of different types depending upon the type of input. Convolution can be 1D, 2D and 3D. One dimensional convolution is applied to the 1D input data. Two-dimensional convolution is applied on the images which has two dimensions (x and y). Three-dimensional convolution is applied to the coloured RGB images. [21]

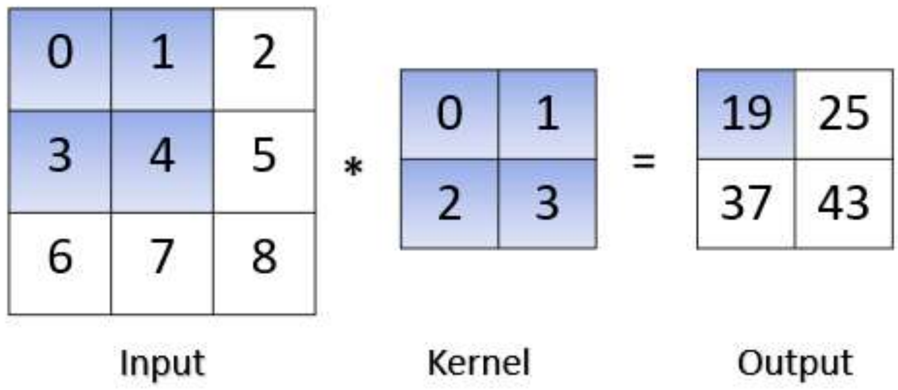


Fig.3.4 Convolution (can be a matrix of pixels).

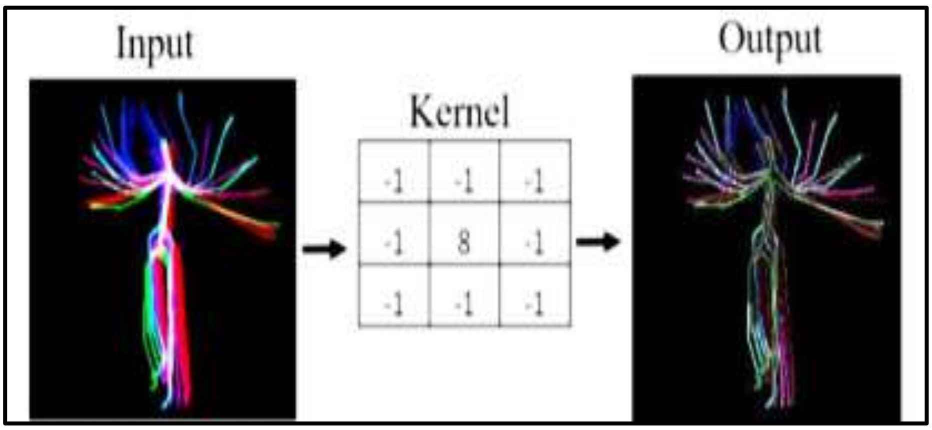


Fig.3.5 Convolution Using Edge Detection Kernel

Pooling operation:

Pooling operation is the second layer in the CNN network. The function of pooling is to decrease the matrix dimension given as input with the help of different operations. Generally, it lowers down the dimension of convolved feature map which gets as an output in case of convolution.

Pooling operation can be done in the following ways:

1. **Max pool:**

Image is represented in the form of pixels. We decide on a particular filter and apply it to an input image, Max pooling operation selects the maximum value from that and represents that maximum value as one pixel in output. As we can see in Fig 3.6, we apply a filter of 2x2 on the input image and get the maximum value from that matrix. Also, the max-pooling function of filter 2x2 and stride of (2,2) on the 4x4 matrix gives the reduced dimension matrix of 2x2.

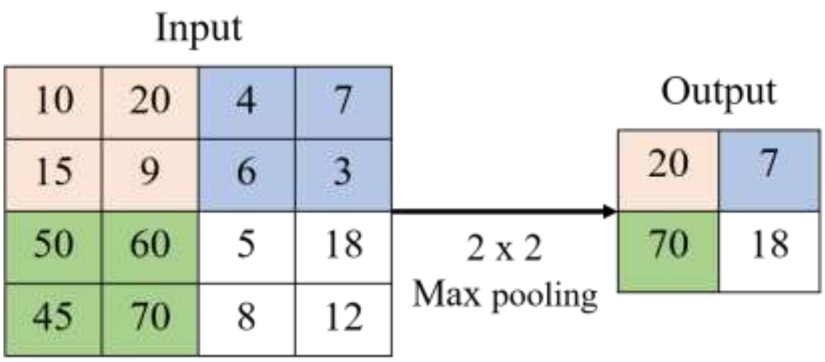


Fig.3.6 Max-pooling Operation

2. **Average pool:**

Average pooling operation finds the average of that particular patch in consideration. In Fig. 3.7, the average of 5,9,7 and 3 is 6. So, the 2x2 patch is downsampled to its average value and the 4 x 4 image is reduced to th 2 x 2 image.

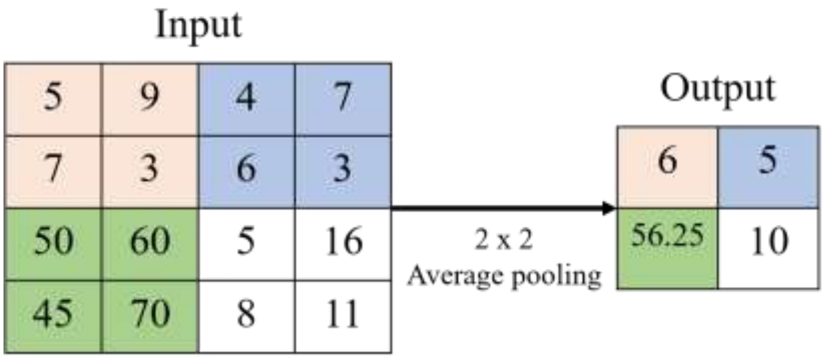


Fig. 3.7 Average pooling operation.

3. **Sum pool:**

Sum pooling operation calculates the sum of all the pixel values in the patch and converts them into its sum value. In Fig. 3.8, patch with 4,7,6 and 3 is converted to their summation i.e. 20.

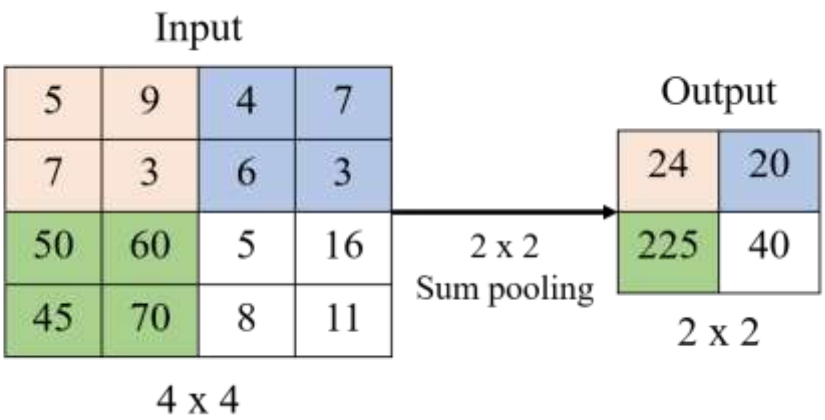


Fig. 3.8 Sum pooling operation.

Output Layer or Dense-Layer:

The dense layer can also be termed as the Fully-Connected-Layer(FCC), this layer is placed at the end of the CNN network. The input of FCC is the last convolved feature map image. The function of this layer is to flatten the image and convert all the pixels into a 1-D array. After that, multiple hidden layers of a neural network are applied to learn the patterns. In a neural network, neurons are interconnected with each other and on each neuron, one activation function is applied. Fig 3.9 shows the complete CNN structure for digit recognition.

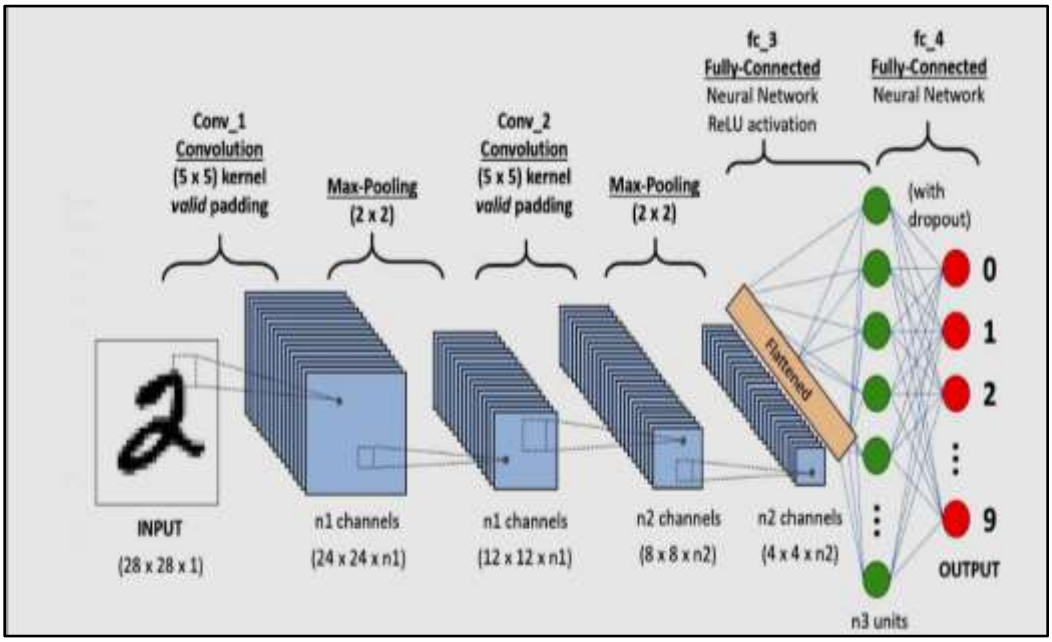


Fig. 3.9 CNN for handwritten digit recognition.

Soft-Max Function:

The soft-max function is an activation function, used at the output of neural networks. It gives the output in the form of the probability of a particular class. For-example-for six different activities it gives output as with how much probability each activity is recognized.

$$\sigma(z)_j = e^{z_j} / \sum_{k=1}^{N} e^{-z_k} \qquad (3.1)$$

Where: j → action

z → output

N → total no. of actions

3.1.4 RECURRENT NEURAL NETWORK (RNN):

RNN is a special type of ANN-Artificial-NeuralNetwork that overcomes the issues faced by the feed-forward-neural-network. RNN algorithm is promising and strong algorithm as it can store the previous information in its internal memory which can easily tackle the temporal behavior of input.

Input to this network is the type of dataset whose input and output are related to each other. Every value of present state input is dependent upon the previous state. It takes sequential or time-series data as an input. Sequential data is the ordered data→ for example a sentence has words in an ordered manner or arranged form. Because of the factor of internal memory, this network can store the output of the prior state (previous hidden state) and alter the present output and hidden state. Due to this reason, R.N.N. is a very trending algorithm in the field of Natural-Language-Processing.

Memory blocks are used to store the information or the value of the previous state. The below Fig 3.10 represents the unfolded version of RNN and we can see that the following parameters are present:

x → input

h → hidden layer

y → output

W_{xh} → weight parameter from input to hidden state

W_{hy} → weight parameter from hidden state to output

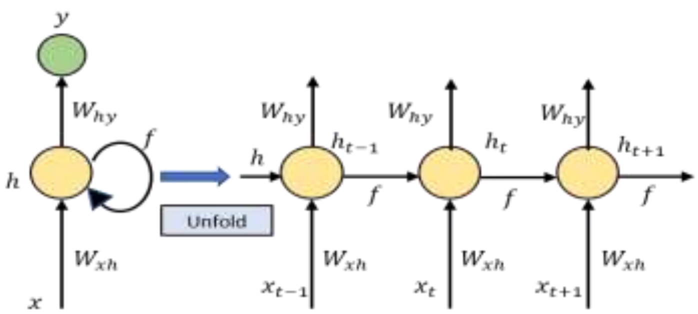


Fig. 3.10. Unfolded version of RNN.

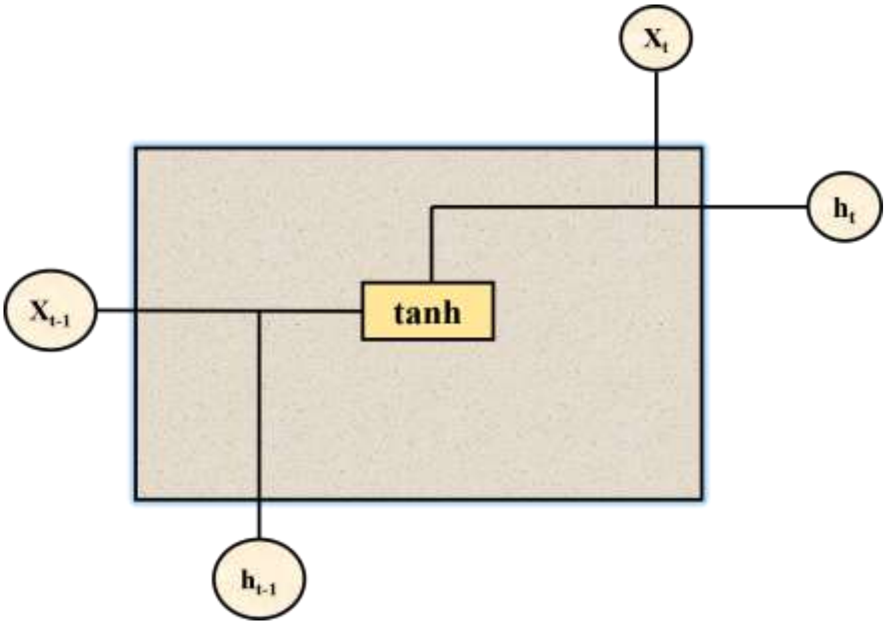


Fig..3.11 Internal structure of RNN

With reference to the above figures → t (present time state), t-1 (previous time state) and t+1 (future time state) is shown. h_{t-1} (hidden state at t-1) is saved in memory and at time t, it acts as an input. [36]

The inputs present at time t are:

x_t → "current input"

h_{t-1} → "previous hidden state"

Output at time t is:

y_t → "current output"

h_t → "current hidden state"

During the internal processing of this function (Fig. 3.11), the activation-function used is hyper-tangent (tanh) function. We can refer to the below equation which shows a mathematical expression of "hidden state" at instant t.

$$h_t = \tanh\,(W_{hh}h_{t-1} + W_{xh}x_t) \qquad (3.2)$$

There are some issues faced by RNN based architecture:

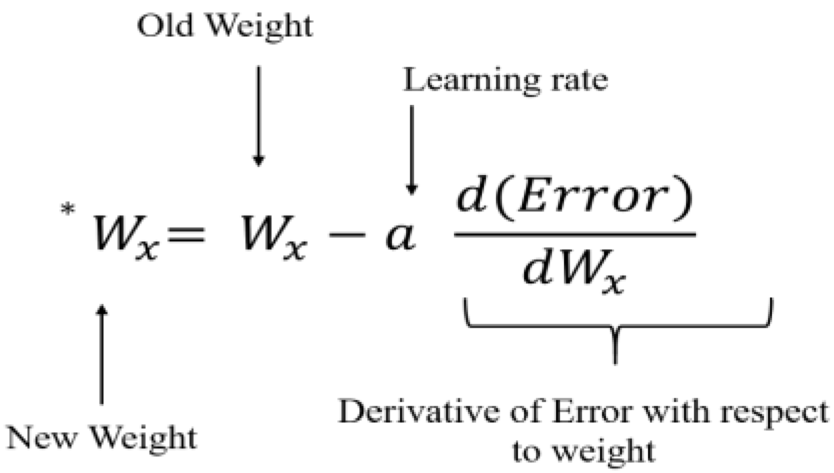
1. Exploding Gradient

2. Vanishing Gradient

A gradient could be a partial-derivative with respect to its input. A gradient-measures that what amount of output alters with respect to the input.

Exploding Gradient Problem:

The exploding gradient is a very common problem in deep learning models. This problem occurs during the process of backpropagation while training the model. During the process of training weights of the neural network are updated. Weight updating is based upon the old weight, learning rate and the gradient of error. Exploding gradient problem occurs when the gradient value (Derivative of Error with respect to weight) becomes too large and the difference between the old weight and new updated weight also becomes large. Due to this model didn't get the global minimum and the model becomes very unstable. Eventually, model will not able to continue training and represent the weight values as NaN.



Weight Update Equation

Vanishing Gradients Problem:

The vanishing gradient problem is another issue faced by RNN during the backpropagation of the model. This problem is also related to the weight update equation. This occurs when the gradient term in the weight updating equation becomes very low or close to zero. Due to this, the new updated value of weight is approximately equal to the old weight value. This stops the training of the model.

These problems can be solved by using the LSTM and GRU architecture.

- **LONG SHORT-TERM MEMORY(LSTM):**

LSTM networks are type of RNN, having a memory to store information. LSTM-network has the ability to handle the issues that we came across in the case of RNN-network. LSTM handles the problem of vanishing gradient.

RNN suffers from the issue of short-term memory. RNN can only store the information of just the previous state but if we want the information from the third previous state, RNN cannot able to do it. This concept where the information of previous states is required is used majorly in NLP use cases. If we talk about the LSTM network, it can easily handle long-term dependencies. In RNN, during backpropagation, it suffers from a problem of vanishing gradient. Gradient value gets shrinked during backpropagation through time and not able to contribute in the learning of a model. LSTM can easily handle this issue of vanishing gradient.

The internal architecture of this network contains the memory-blocks and further blocks has the memory-cells. Moreover, inside the network, different gates are present.

These gates are responsible for the flow of information, they can regulate the information flow through the network. These gates are forget-gate, input-gate and output-gate.[36]

- **Forget gate→** Forget-gate neglects the information which is not relevant and pass the information which is required to transfer to cell state.

- **Input gate→** Input gate allows the new information to pass through cell state.

- **Output gate→** Output gate use the hyper tangent function to give the filtered result.

Working of LSTM network:

In foremost step, current input and previous hidden state get summed up (concatenate). This concatenated output is fed to the forget gate. After entering into forget gate sigmoid function is applied. If the output of sigmoid function is plus one(+1), information is allowed to enter the cell state. If the output of sigmoid function is minus one(-1), then information is not allowed to enter the cell state. So, the function of forget layer is to remove the non-relevant data.

The concatenation of the hidden-state and the input is also fed to the input-gate and the output gate. The 'input gate' controls which new information is allowed to pass through the cell state. Output gate gives filtered output(using tanh function). Pointwise multiplication of 'output' and the 'new-cell-state' results in new hidden state.

$$h_t = \sigma(W_{i,h} \, . \, x_t + W_{h,h} \, . \, h_{t-1} + b) \tag{3.3}$$

where:

W: weight parameter

h_{t-1}: previous hidden state

b: bias

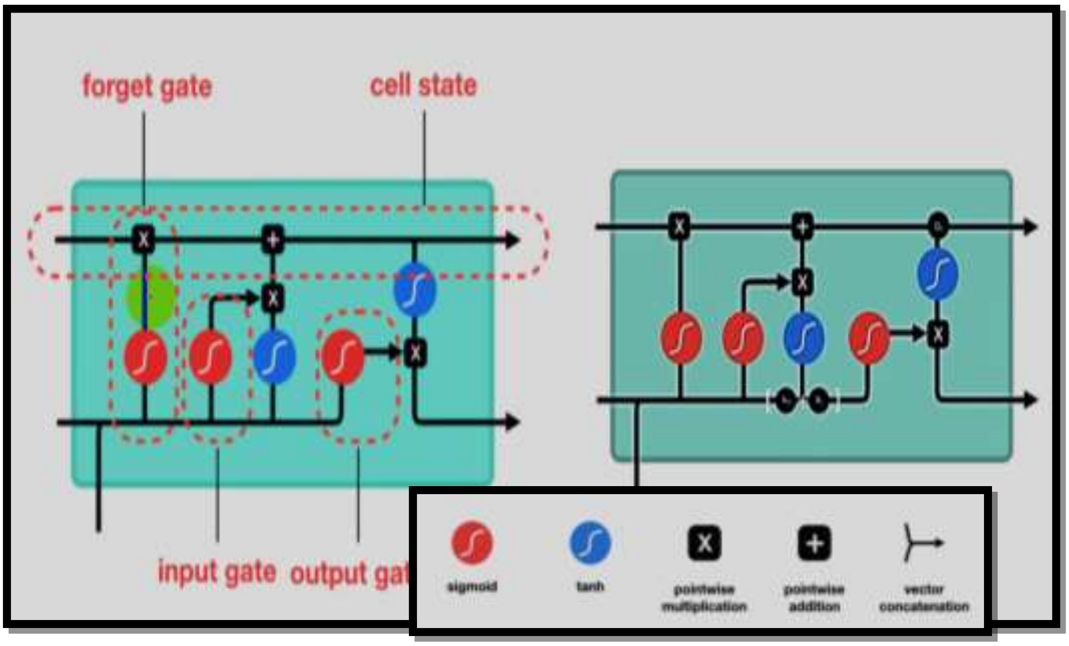


Fig.3.12 Different gates in LSTM

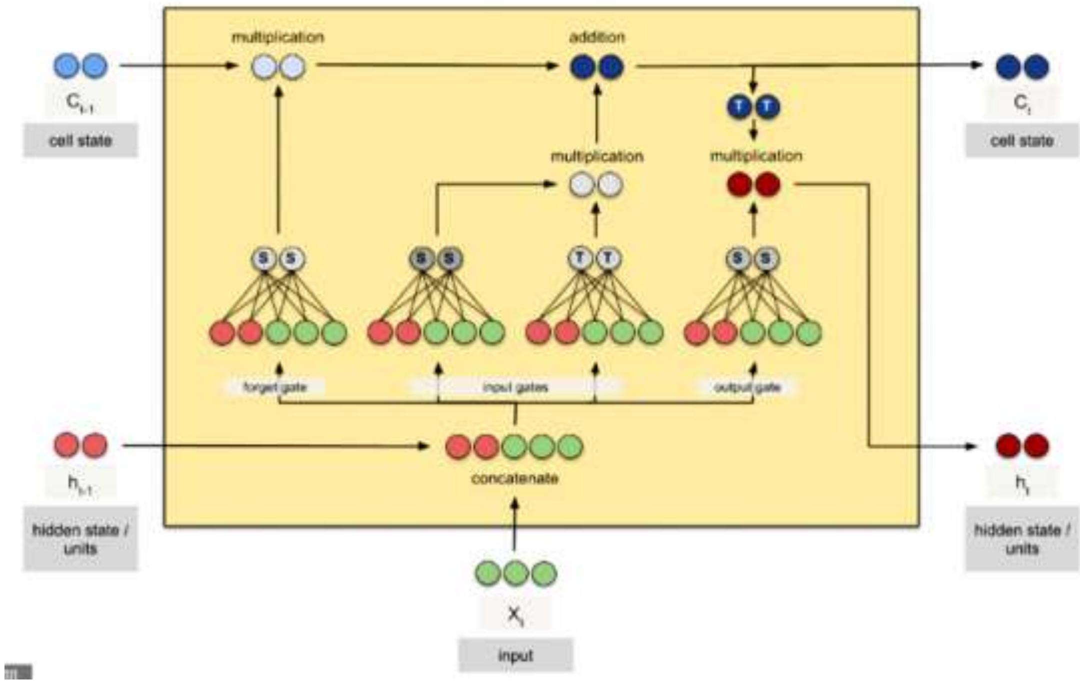


Fig.3.13 Internal flow of LSTM

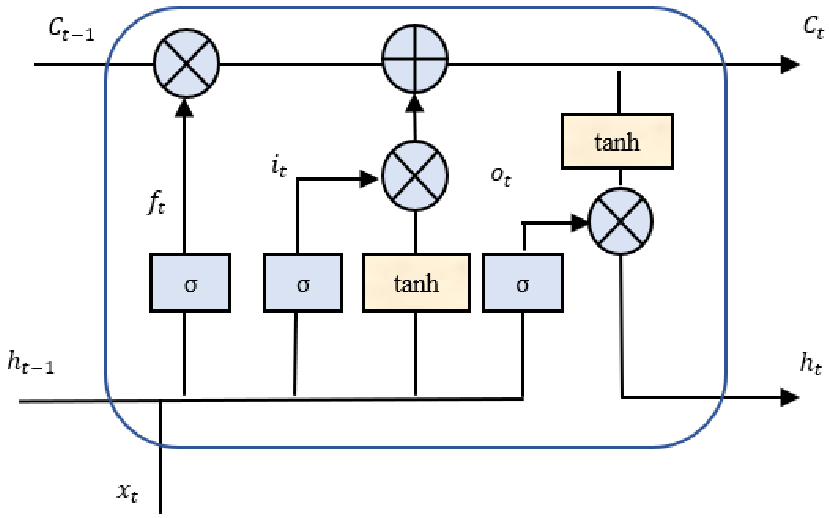


Fig. 3.14. Internal architecture of LSTM.

$$i_t = \sigma(W_{hi}h_{t-1} + W_{xi}x_t + b_i) \tag{3.4}$$

$$f_t = \sigma\left(W_{hf}h_{t-1} + W_{xf}x_t + b_f\right) \tag{3.5}$$

$$o_t = \sigma(W_{ho}h_{t-1} + W_{xo}x_t + b_o) \tag{3.6}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \tag{3.7}$$

$$h_t = o_t \odot c_{t-1} + \tanh(c_t) \tag{3.8}$$

where σ is the 'logistic-sigmoid-function', $\odot$ represent point wise multiplication and here, i_t, o_t, f_t, and c_t are 'input-gate', 'forget-gate', 'memory-cell' and 'output-gate' 'vector-functions(activation functions)' respectively. b_i, b_o, b_f and b_c are the 'bias-terms' and W_{ab} represents the 'weight-matrix' from *a* to *b*. [21]

3.1.5 DIFFERENCE BETWEEN RNN AND ANN:

The below table demonstrate the small comparison between the "Feed-Forward-Neural-Network" and "Recurrent-Neural-Network".

TABLE 3.1 COMPARISON BETWEEN ANN AND RNN

ANN	RNN
No memory	Memory
Different weights at each layer	Share same weights on every layer
Backpropagation algorithm	Backpropagation through time

There are some parameters based on which we can compare these two networks.

- The first and foremost factor is a memory which distinguishes NN from RNN. As RNN can store the previous inputs to alter the current output state. Whereas, feed-forward-neural-network doesn't have any memory, so it cannot able to store the previous inputs.
- Another main distinguishable factor is that RNN shares the equal weights within each layer of the network while feed-forward neural-network have different weights across each layer.
- To find the derivative or gradient, RNN uses an algorithm→ Back-Propagation-Through-Time (BPTT). Whereas feed-forward-neural-network uses a normal backpropagation algorithm. In BPTT, errors got summed up at each time step.

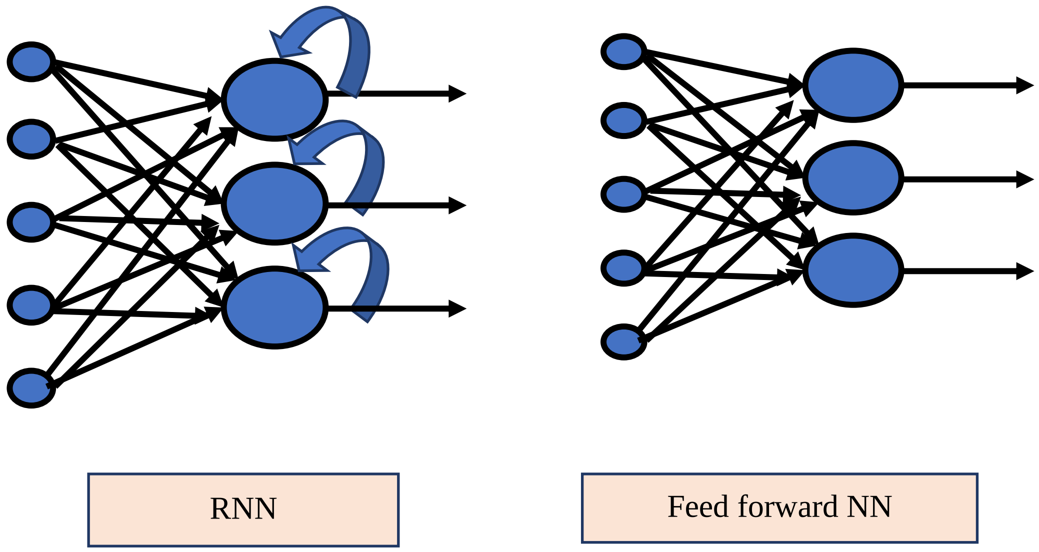


Fig. 3.15 RNN and NN

- In feed-forward neural-network, the flow of information is only in one direction i.e., from input to output (input-layer→hidden-layer→output-layer). It cannot able to remember anything as it depends only on the present-input.
- In RNN, the flow of information is from input to hidden layer then to output and then again, this output is feed to the input of the next time state. We can refer to the above figure.

3.1.6 TYPES OF RNN

- One to many
- Many to one
- Many to many

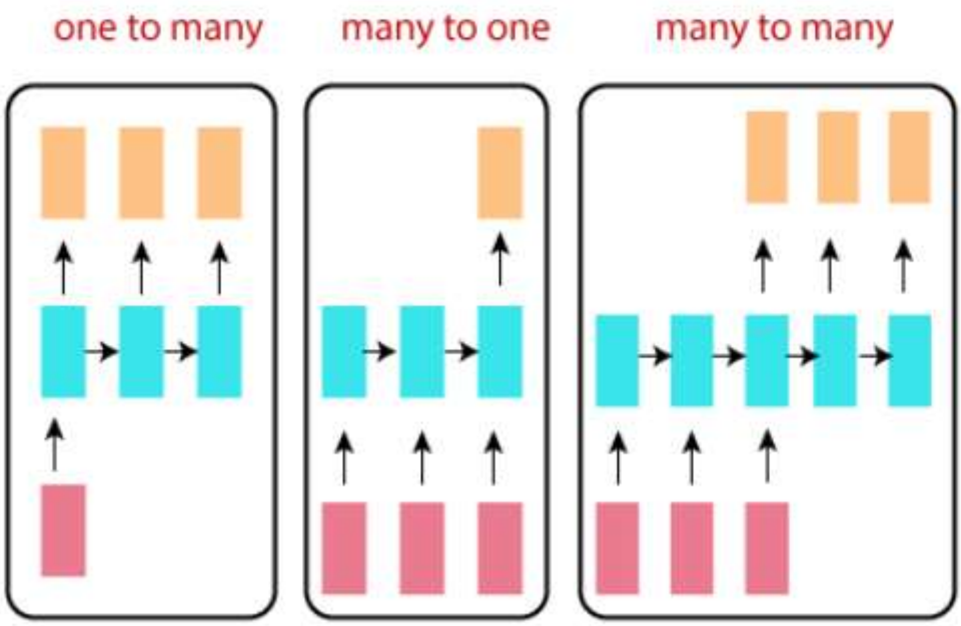


Fig. 3.16 Types of RNN

One to many: In one-to-many RNN, the input is of fixed size and the output is a sequence of data. E.g.: “Image captioning”- during the process of image captioning input is a single image and output is the sequence of words.

Many to one: In many to one RNN structure, Input → sequence of information and Output → fixed length. For example- Sentiment analysis, take the sequence of words as input and analyse its sentiment and return as fixed output i.e., positive or negative.

Many to many: In many to many RNN structure, both input and out are the sequence of information. For example- Machine translation: The sequence of input is given to the machine in one language which converts it into another language and provides output as a sequence of data. [37]

CHAPTER 4

PROPOSED WORK

4.1. DATASET USED

WISDM: Wireless Sensor Data-Mining

In our research work, we have used WISDM dataset and it is available on its official website. The data in this dataset is gathered by two sensors that are widely used for HAR i.e., accelerometer and gyroscope sensors. These sensors are inbuild into smartphones. Smartphones sense the location details in the three-dimensional form i.e., x, y and z.

Dataset has a total of 6 attributes (activities) which are carried out by individuals performing these 6 activities at different places and different periods. These activities are coming Downstairs(down), going Upstairs(up), Standing(std), Sitting(sit), Jogging (jog) and Walking(walk). All activities are not equally divided.

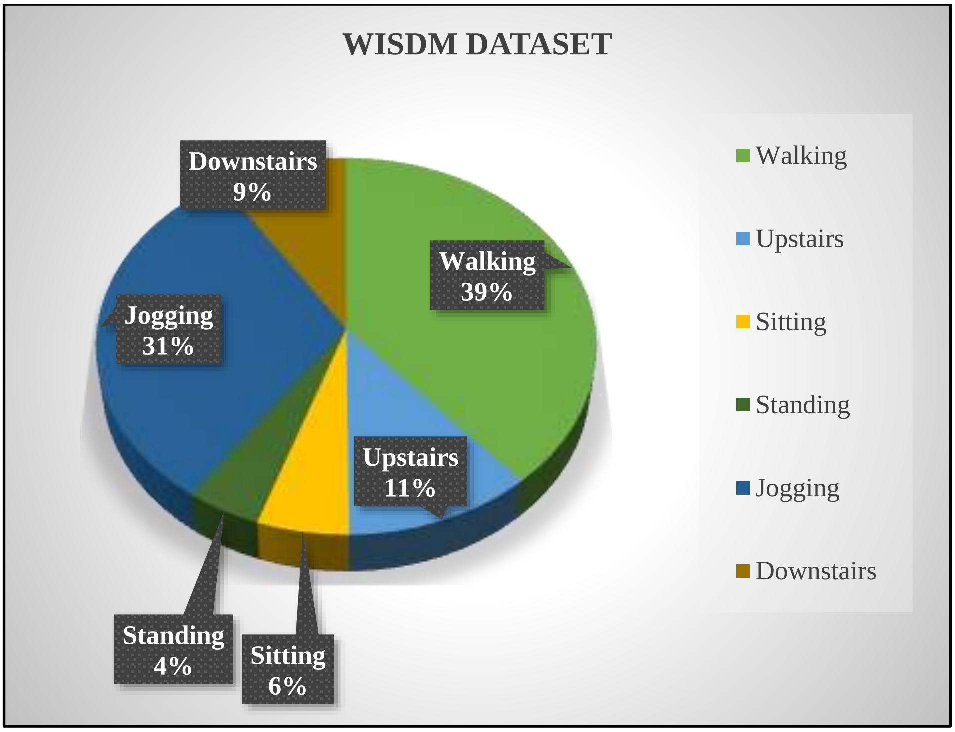


Fig. 4.1 Distribution of activities in WISDM Dataset

Signals are plotted for a particular dimension i.e., x, y & z and for each activity. Therefore 3 signals for each activity and 18 signals for 6 activities.

Signals were plotted with a sampling-rate 50hz and represented as a ‘time-series’ for each dimension. We have used the Butterworth filter with a frequency of 20hz to remove noise from the signals and to achieve good results. Then the values are ‘normalized’ within the range of (-1,1).

Pre-processing of data set is done to represent it without any noise and convert it into data frame to represent it. Also, the string type is converted into a numeric type to perform further operations on it.

WISDM data is not balanced as the values for the walking feature are most i.e., 38% of whole data and the standing feature vector has the least values i.e., 4.4%. A pie chart in Fig 4.1 is representing the percentage distribution of each activity of the WISDM dataset. [21]

	User ID	Activity	Time	X	Y	Z
0	33	Jogging	49105962326000	-0.6946377	12.680544	0.50395286
1	33	Jogging	49106062271000	5.012288	11.264028	0.95342433
2	33	Jogging	49106112167000	4.903325	10.882658	-0.08172209
3	33	Jogging	49106222305000	-0.61291564	18.496431	3.0237172
4	33	Jogging	49106332290000	-1.1849703	12.108489	7.205164
5	33	Jogging	49106442306000	1.3756552	-2.4925237	-6.510526
6	33	Jogging	49106542312000	-0.61291564	10.56939	5.706926
7	33	Jogging	49106652389000	-0.50395286	13.947236	7.0553403
8	33	Jogging	49106762313000	-8.430995	11.413852	5.134871
9	33	Jogging	49106872299000	0.95342433	1.3756552	1.6480621
10	33	Jogging	49106982315000	-8.19945	19.57244	2.7240696
11	33	Jogging	49107092330000	1.4165162	5.7886477	2.982856

Fig.4.2 Balanced WISDM dataset

The total 'number of sample' values in this dataset is 1,098,207. The activities are unstructured and unbalanced. So, we need to balance the data. We balance the data by making the values of each activity the same as the standing activity i.e., values=4.4% of the dataset.

The snippet of the dataset is shown in Fig 4.2.

There are total of six columns in the dataset. The time column in the dataset shows the time of a smartphone in nanoseconds. Unique id shows the id of a particular person, 36 individuals have collected the dataset each of them has a unique id. We can neglect these features as they are not contributing to the classification of activity. Therefore, we only need four columns i.e., X, Y, Z and Activity.

4.2. PROPOSED MODELS:

In our research work, we have proposed two classification models to recognize human activities. The proposed models have shown accurate results. Moreover, the complexity of the network architecture is reduced in terms of the number of trainable parameters.

A. CNN Model:

The first model is based upon the CNN architecture. CNN network has the capability to extract the features and optimize itself while learning. Also, CNN classifies the input. As explained and discussed in Chapter 3 about the CNN, it has usually 3 layers 'convolution layer', 'pooling layer' and 'output layer'.

In our proposed CNN model, we have implemented four CNN layers with different values of kernel-size. Kernel-size decides the depth of feature-extraction, the larger the kernel size the deeper features are extracted.

We have ReLU activation function in between the hidden-layers of the neural network. ReLU is "Rectified-Linear-Unit", this activation function converts input to positive linear value. The mathematical-expression of ReLU function is shown is in equation (4.1).

$$\left.\begin{aligned} f(s) &= 0; \rightarrow for\ s < 0 \\ f(s) &= s\ ; \rightarrow for\ s < 0 \end{aligned}\right\} \quad (4.1)$$

If we use a large kernel size, this results in deeper extraction of features but also it gives rise to a problem of overfitting. Overfitting of a model leads to the increase in generalization error which means our model is not generalized and the model will not able to perform accurately on new data or test data. To reduce the overfitting effect on the model we have added 'dropout-layers' in between the CNN layers. The dropout-layer drops off the neurons so that they will not able to learn all the features. This will add a regularization effect to the model.

The output of CNN contains the fully-connected-layer and the Soft-max layer. The fully-connected layer has two hidden-layers with 32 neurons and 64 neurons. The Soft-max layer contains 6 neurons which is representing the number of outputs.

B. LSTM Model:

Second model is based upon the LSTM architecture. We have implemented two LSTM layers with 32 neurons in each layer. LSTM is a type of network which has memory and can able to handle the time-series input. Also, LSTM network is capable to handle the problem of long term dependencies and vanishing gradient.

Batch normalization layers are used in between the LSTM layer. BN-layer normalizes the input sequence during the time of training of a model and also gives the stabilized output. Dropout layers are added to give the regularization effect which reduces the problem of overfitting in a model.

The proposed model is very promising and giving very good-results in terms of 'accuracy'. Also, the complexity of the model is very low in comparison with other SOTA models. The complexity of the model is defined in terms of a number of trainable parameters. TABLE 4.1 shows the number of trainable parameters when we have used one layer of

LSTM and when we have used two layers of the LSTM model. Further, if we are increasing the layers of LSTM the accuracy is not increasing but the ‘complexity’ of the model is increasing due to the increased number-of-weights during training. [21]

TABLE 4.1. TOTAL PARAMETERS

No.-of layers:	Total-Parameters
One-LSTM-layer	13,622
Two-LSTM-layers	21,922

CHAPTER 5

EXPERIMENTAL RESULTS

In our research experiment, the WISDM-dataset is used which is a public dataset. This dataset is about the different human activities gathered using smartphone sensors. We have used Metadata for our experiment. We have pre-processed the dataset and make it ready for further processing. The dataset is splited into a “training set” and the “test set” with a ratio of 7:3. The software which I have used is Python with other important libraries like Keras with TensorFlow 2.0, Numpy, Pandas, Matplotlib, etc. Initial values of weight and biases are the default values are “TensorFlow” values. The error function(loss) we have used is the cross-entropy loss(error) function because it results in the faster-convergence of the network model. Loss function helps to find the error in our learned network i.e. difference between “true” and “predicted” value.

Optimizers are used to optimize the errors (loss) that we get. There are different optimizer functions used in TensorFlow. We have used Adam Optimizer; it is a combination of “AdaGrad” and “RMSprop” optimizers and is known as “Adaptive Moment Estimation”. The advantages of Adam Optimizer are that it uses less memory and is efficient. As discussed earlier, we have evaluated two models; the first model is the CNN model and the second model is LSTM model. CNN model and LSTM model is trained for 100 and 200 epochs respectively. Accuracy and loss of the model are calculated by altering the parameters like ‘batch size’, ‘dropout layer’, and ‘no. of epochs ’.

The number of epochs was selected according to the convergence of the model. If below one particular value of epoch our model gets converge, there is no profit of increasing the no. of an epoch. Our model is performing very well in terms of ‘Accuracy’ and ‘F1 score’ without suffering from the issue of ‘overfitting’. For our CNN model, the best-selected batch-size is 64 and the dropout-value is 0.5 with 100 epochs. For the LSTM model, the best-selected batch size is 64, and the dropout value is 0.5 with 200 epochs. The batch size of 64 is the best selected optimal size as when we increase the batch size it results in slow convergence of model and issue of overfitting which affects the model performance. When we reduce the batch-size, it results in the fast convergence of the model but unable to locate the exact global minima. Also, reducing the batch-size results in ‘underfitting’ of the model and it behaves noisy and unstable. For ‘CNN’ model. we obtained an ‘Accuracy’ of 95% and for the ‘LSTM’ model obtained 97.5%.

Experimental graphs and confusion-matrix of both the models are shown below.

Fig. 5.1 represents the confusion matrix of the CNN model. The confusion matrix shows the performance of the classifier or the classification model implemented. We can observe from Fig 5.1 (Confusion-matrix of CNN model) that we get confused when it tries to classify the between standing and walking activity.

Fig 5.2 represents the accuracy curve and Fig 5.3 represents the loss curve of the CNN model for ‘training’ and ‘validation-set’. It can be observed from the accuracy curve (Fig 5.2) that the CNN model gets converged between 30 to 40 epochs. In the loss curve, the validation-loss is slightly on higher side but can be neglected because convergence is smooth.

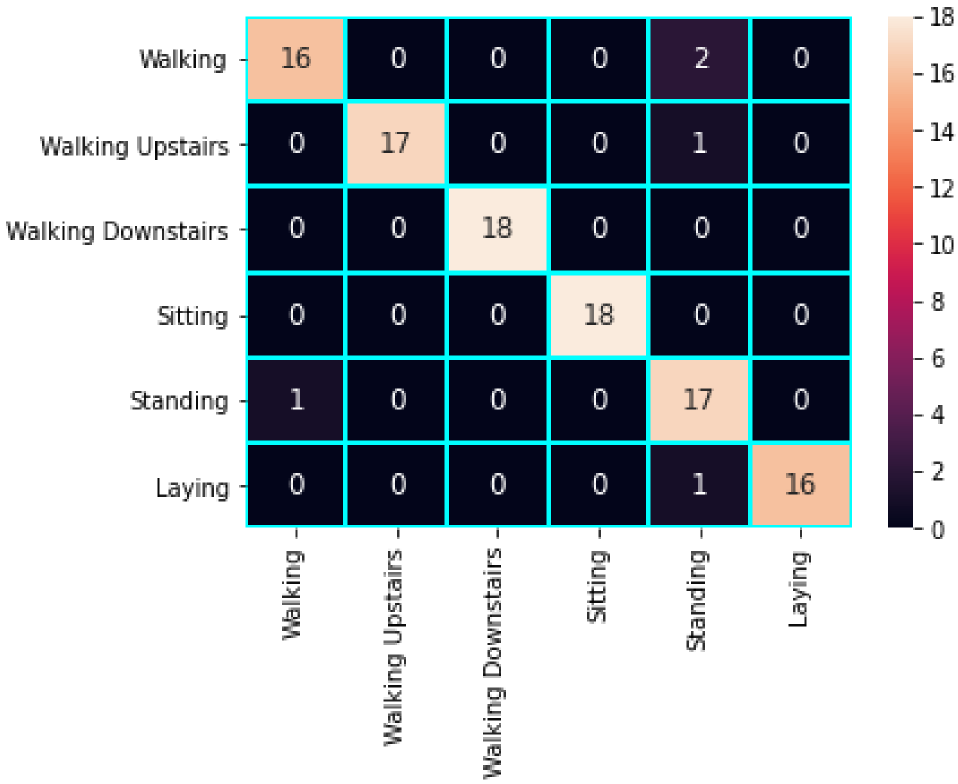


Fig. 5.1 Classification Confusion-Matrix for CNN model.

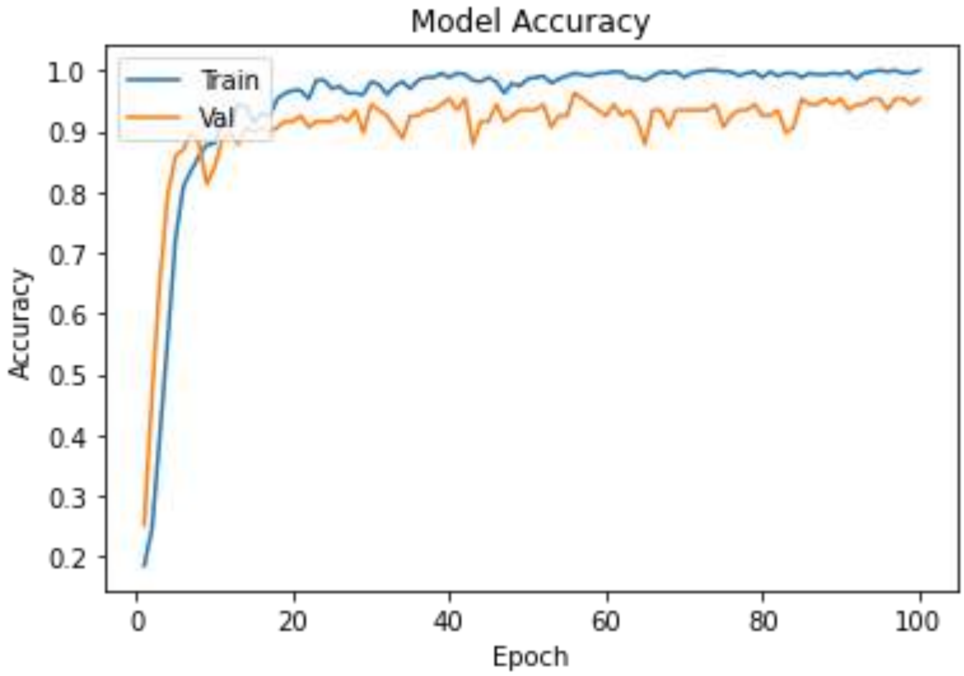


Fig 5.2 CNN-Model Accuracy curve

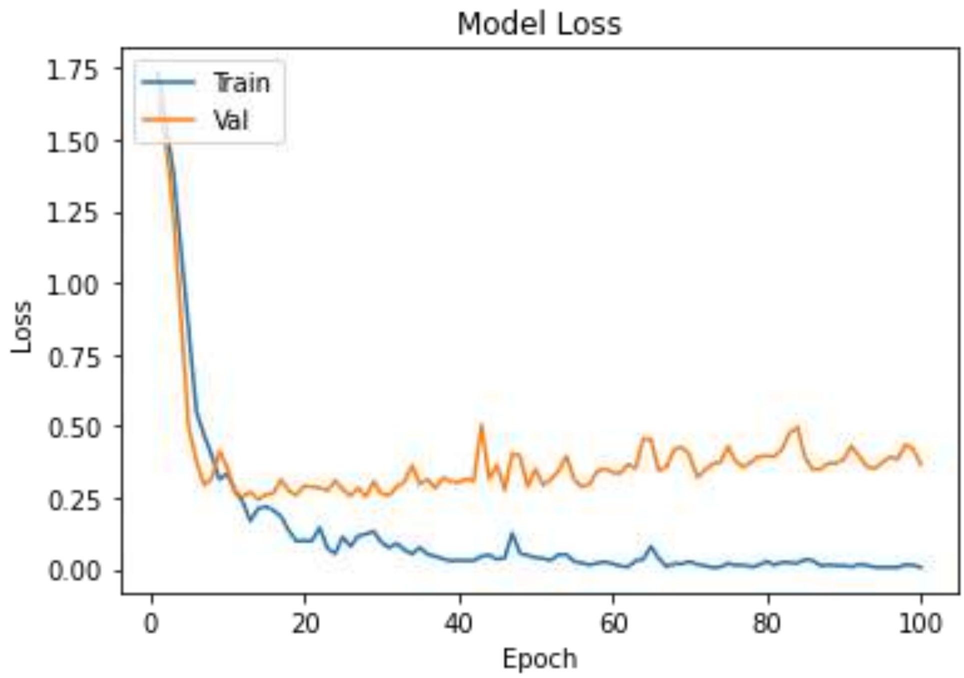


Fig 5.3 CNN-Model Loss curve

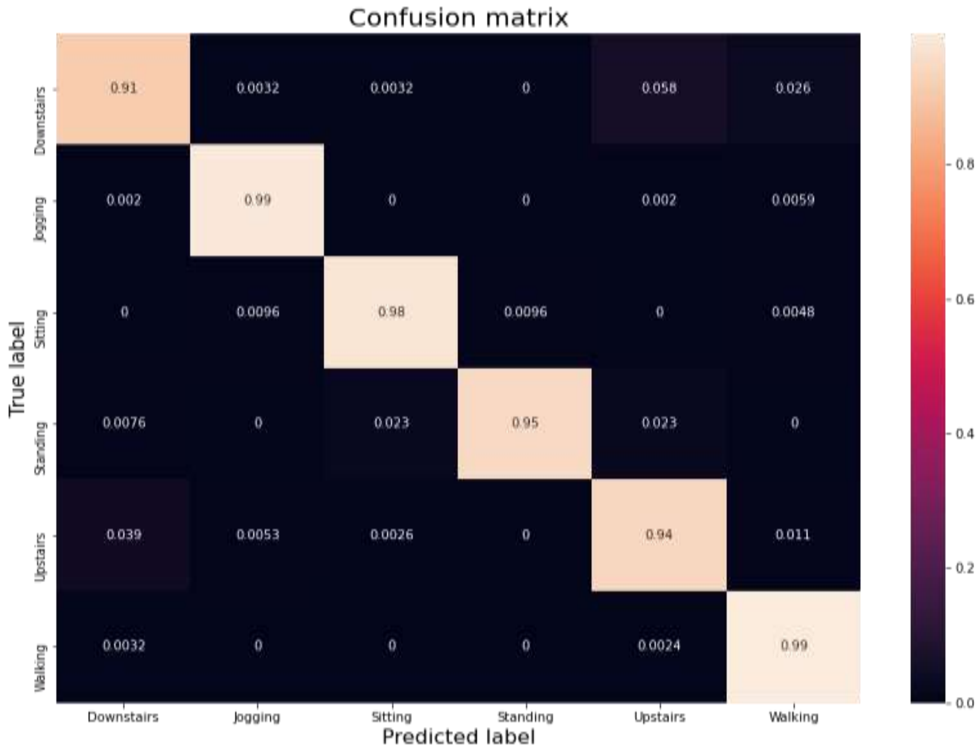


Fig. 5.4 Classification Confusion-Matrix for LSTM model

Fig. 5.5. LSTM Model Accuracy

Fig. 5.4 represents the confusion matrix for the LSTM-model. The confusion-matrix shows that the model gets confused when tries to classify between ‘going downstairs’ and ‘coming upstairs’. The accuracy curve and the loss curve for the LSTM-model are shown in Fig. 5.5 and Fig. 5.6 respectively. It can be observed from the accuracy curve that the model is converging after 75 epochs. the experimental results are quite promising when compared with deep-learning models implemented with same dataset.

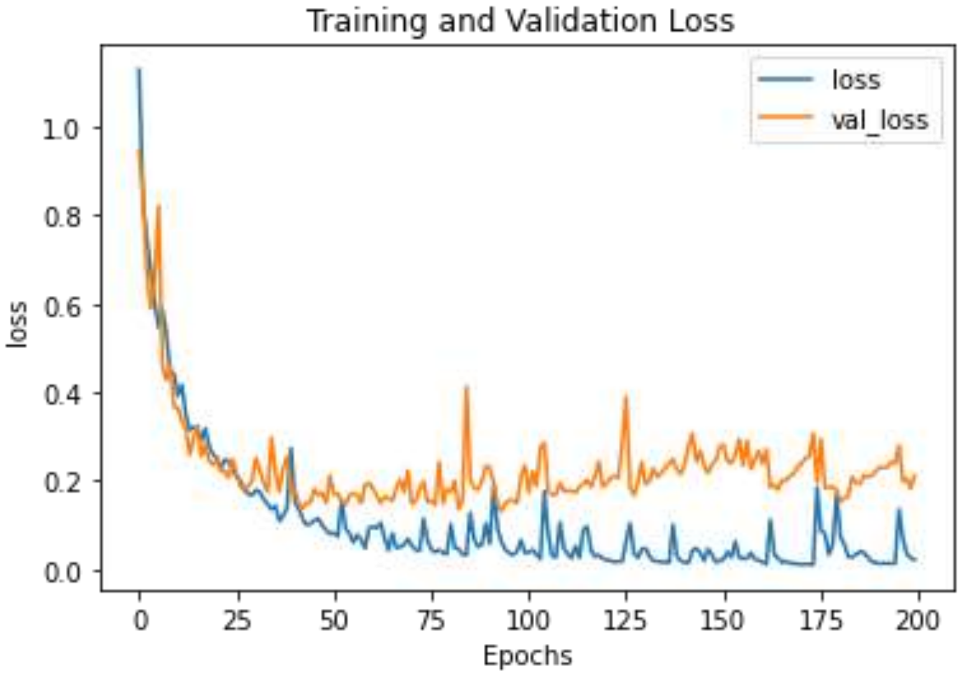


Fig. 5.6. LSTM Model Loss

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

Human Activity Recognition(HAR) is the leading area of research for different applications like in “Health and Medical”, “Transportation”, “Security and Surveillance”, “Robotics”, “Entertainment”, etc. In this thesis, we proposed a Smartphone-based ‘Human Activity Recognition system’. The six different human activities are classified. The proposed model is evaluated on the WISDM dataset which includes time series data, collected using sensors inbuilt in smartphones.

The two models were evaluated and validated on the test and train data. ‘CNN classifier’ and ‘LSTM network’ are used to classify six human-activities. The proposed model has low loss and high accuracy. The complexity of the proposed-architecture is low because we have used only two ‘LSTM-layers’. Moreover, different parameters (‘batchsize’, ‘dropout’ and ‘no. of epochs’) are altered to achieve good results. We have achieved ‘Accuracy’: up to 97.5% using the proposed ‘LSTM-network’. It is found that the proposed “LSTM model” is more accurate in comparison to other state-of the art models available in the open-literature.

For future scope, we can add more sets of complex activities to further improve our model. Different deep learning techniques can be fused and applied to different datasets. Also, more robust features can be extracted which can efficiently classify the activities. The challenging task is to classify the complex human activities in a real-time scenario which can be a good future scope for research. To make a generalized and robust model for the HAR system, we need to create our dataset with the help of smartphone sensors. Using different types of sensors embedded in smartphones leads to more consumption of battery which can be a constraint. So, advancement in battery optimization techniques can improve the overall HAR system.

REFERENCES:

[1] Cho Nilar Phyo,Thi Thi Zin , Pyke Tin, “Deep Learning for Recognizing Human Activities Using Motions of Skeletal Joints”, IEEE Transactions on Consumer Electronics , Volume: 65, Issue: 2, May 2019

[2] A. Avci, S. Bosch, M. Marin-Perianu, R. Marin-Perianu, P. Havinga, “Activity recognition using inertial sensing for healthcare, wellbeing and sports applications: A survey,” International Conference on Architecture of Computing Systems, 2010.

[3] W. Lin, M.T. Sun, R. Poovandran, Z. Zhang, “Human activity recognition for video surveillance,” in: IEEE International Symposium on Circuits and Systems, 2008.

[4] T. Liu, F. Bingfei, and L. Qingguo, “The invention relates to a wearable motion sensor and a method for resisting magnetic field interference,” 2017.

[5] K. Yatani and K. N. Truong, “BodyScope : A Wearable Acoustic Sensor for Activity Recognition,” Ubicomp’12, pp. 341–350, 2012.

[6] H. Lu, W. Pan, N. D. Lane, T. Choudhury, and A. T. Campbell, “Soundsense: Scalable sound sensing for people-centric applications on mobile phones,” in Proc.of the 7th Int’l Conf. on Mobile Systems, Applications, and Services. ACM, pp. 165–178, 2009.

[7] A. Yazar, A. Cetin, and B. Toreyin, “Human activity classification using vibration and PIR sensors,” in Signal Processing and Communications Applications Conf. (SIU), pp. 1–4, April 2012.

[8] B. Mirmahboub, S. Samavi, N. Karimi, and S. Shirani, “Automatic monocular system for human fall detection based on variations in silhouette area,” IEEE Trans. on Biomed. Eng., vol. 60, no. 2, pp. 427–436, 2013.

[9] L. Bao and S.S. Intille, “Activity Recognition from User-Annotated Acceleration Data,” In: Ferscha A., Mattern F. (eds) Pervasive Comput., LNCS, vol. 3001, 2004.

[10] A. Jain and V. Kanhangad, “Human activity classification in smartphones using accelerometer and gyroscope sensors,” IEEE Sensors J., vol. 18, no. 3, pp. 1169–1177, Feb. 2018.

[11] P. Gupta and T. Dallas, ‘‘Feature selection and activity recognition system using a single triaxial accelerometer,’’ IEEE Trans. Biomed. Eng., vol. 61, no. 6, pp. 1780–1786, Jun. 2014.

[12] E. Fullerton, B. Heller, and M. Munoz-Organero, ‘‘Recognizing human activity in free-living using multiple body-worn accelerometers,’’ IEEE Sensors J., vol. 17, no. 16, pp. 5290–5297, Aug. 2017.

[13] J. Margarito, R. Helaoui, and A. M. Bianchi, “User-independent recognition of sports activities from a single wrist-worn accelerometer: A template-matching-based approach,” IEEE Trans. Biomed. Eng., vol. 63, no. 4, pp. 788–796, Apr. 2016.

[14] J. Mantyjarvi, J. Himberg and T. Seppanen, “Recognizing human motion with multiple acceleration sensors,” in Proc. IEEE International Conference on Systems, Man and Cybernetics, vol. 2, pp. 747-752, 2001.

[15] Y. Zheng, Q. Liu, and E. Chen, “Time series classification using multichannels deep convolutional neural networks,” in Proc. Int. Conf. Web-Age Inf. Manage. Cham, Switzerland: Springer, 2014, pp. 298– 310.

[16] Y. Chen and Y. Xue, “A Deep Learning approach to Human Activity Recognition based on Single Accelerometer,”IEEE, International Conference on Systems, Man, and Cybernetics, SMC 2015, pp. 1488–1492, Hong Kong, China, October 2016.

[17] C. A. Ronao and S.-B. Cho, “Human activity recognition with smartphone sensors using deep learning neural networks,” Expert Systems with Applications, vol. 59, pp. 235–244, 2016.

[18] F. Ordóñez and D. Roggen, “Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition,” Sensors, vol. 16, no. 1, p. 115, 2016.

[19] A. Murad and J. Y. Pyun, “Deep recurrent neural networks for human activity recognition,” Sensors (Switzerland), vol. 17, p. 11, 2017.

[20] Y. Guan and T. Plotz, "Ensembles of deep LSTM learners for ¨ activity recognition using wearables," Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, vol. 1, no. 2, pp. 1–28, 2017.

[21] L. Dhammi, P. Tewari, "Classification of Human Activities using data captured through a Smartphone using Deep Learning Techniques," IEEE International Conference on Signal Processing and Communication, May 2021

[22] Godwin Ogbuabor, Robert La, "Human Activity Recognition for Healthcare using Smartphones," ICMLC, Macau, China, Association for Computing Machinery, ISBN 978-1-4503-6353-2/18/02, 2018

[23] Kun Xia , Jianguang Huang , and Hanyu Wang, "LSTM-CNN Architecture for Human Activity Recognition", IEEE Access, vol.8,2020.

[24] M. M. Hassan, Md. Z Uddin, A. Md, A. Almogren, "A robust human activity recognition system using smartphone sensors and deep learning," Future Generation Computer Systems 81, 307–313, 2018.

[25] J.R. Kwapisz, G.M. Weiss, S.A. Moore, "Activity recognition using cell phone accelerometers," Sensor KDD, Washington, DC, USA, July 25, 2010.

[26] K.G.M. Quispe, W.S. Lima, D.M. Batista, E Souto, "MBOSS: A symbolic representation of human activity recognition using mobile sensors," Sensors, 18, 4354; doi:10.3390/ s18124354,2018.

[27] D. Micucci, M. Mobilio, P. Napoletano, "UniMiB SHAR: A Dataset for Human Activity Recognition using Acceleration Data from Smartphones." Appl. Sci. 2017, 7, 1101; doi:10.3390/app7101101, 2017.

[28] M. Panhwar, S. Shah, M. Syed, et al., "Smart Phone Based Fall Detection using Auto Regression Modeling in a Non-Restrictive Setting," Indian Journal of Science and Technology, Vol 10 (5), February 2017.

[29] J. Xu, Z. He, Y. Zhang, "CNN-LSTM Combined Network for IoT Enabled Fall Detection Applications," IOP Conf. Series: Journal of Physics, 2019.

[30] Z. Zheng, J. Du, L. Sun, M. Huo, Y. Chen, "TASG: An Augmented Classification Method for Impersonal HAR," Hindawi Mobile Information Systems, Volume 2018.

[31] T. Sztyler, H. Stuckenschmidt, “On-body Localization of Wearable Devices: An Investigation of Position-Aware Activity Recognition,” IEEE International Conference on Pervasive Computing and Communications (PerCom), IEEE, 2016.

[32] https://www.maximintegrated.com/en/design/technical-documents/app-notes/5/5830.html#:~:text=Accelerometer%20Versus%20Gyroscope&text=Accelerometers%20measure%20linear%20acceleration%20(specified,mV%2Fdeg%2Fs).&text=Then%20we%20can%20measure%20Coriolis,the%20inner%20frame%20and%20substrate.

[33] https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/

[34] https://www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.

[35] https://www.analyticsvidhya.com/blog/2021/04/simple-understanding-and-implementation-of-knn-algorithm/

[36] https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21

[37] https://iq.opengenus.org/types-of-rnn/

LIST OF PUBLICATIONS

[1] Lokesh Dhammi, Piyush Tewari **"Classification of Human Activities using data captured through a Smartphone using Deep Learning Techniques,"** IEEE International Conference on Signal Processing and Communication, May 2021.