Journal of Engineering Science and Technology
Vol. 17, No. 4 (2022) 2315 - 2335
© School of Engineering, Taylor's University

THE CUSTOMIZED 1D CNN FOR SENSOR-BASED HUMAN ACTIVITY RECOGNITION USING VARIOUS BENCHMARK DATASETS

SHILPA ANKALAKI[1,*], M. N. THIPPESWAMY[2]

[1]Department of CSE, Nitte Meenakshi Institute of Technology,
Bangalore, Karnataka, India - 560064
[2]Department of CSE (AI&ML) and Department of CSE (Cyber Security),
Ramaiah Institute of Technology, Bangalore, Karnataka, India – 560054
*Corresponding Author: shilpaa336@gmail.com

Abstract

Rapid development in detection and recognition of human activity (HAR) based on wearable sensor data has been witnessed in the recent past and it has become one of the significant research fields because of its advanced applications in domains such as healthcare, athletics, assistance, and rehabilitation. In recent years, activity recognition has been accomplished using wearable sensors like accelerometers, gyroscopes, magnetometers, etc. In traditional approaches, feature engineering has to be performed by applying some heuristic methods in order to determine the type of activity. The process of feature engineering requires domain experts to extract efficient and distinct features from the sensor data. Recognition of complex activities, accuracy, and efficiency of the model relies upon the features extracted during the feature engineering process. Critical analysis of this whole process showed that the process of extraction of handcrafted features is very involved and time-consuming when complex recognition problems are handled. In addition, it is quite inefficient and inaccurate. In addition, it is quite inefficient and inaccurate. A novel Customized 1D-CNN (C1DCNN) is proposed in this work in order to remove the tedious process of handcrafted feature extraction and to improve the recognition efficiency and accuracy. Automatic feature extraction is accomplished by Convolutional Neural Networks (CNN). In the proposed scheme C1DCNN, the recognition of human physical activities is accomplished using deep learning techniques it has been evaluated on five datasets namely UCI-HAR, Opportunity, PAMP2, Daphnet Gait, and Sphere datasets. This work proposes a C1DCNN and employs deep learning algorithms like Deep Neural Networks (DNN) and Bi-Directional Long Short-Term Memory (BD-LSTM). The performance of these deep learning approaches is analysed using various activation functions, hidden layers, optimizer techniques, and learning rates. The performance evaluation is based on Precision, Recall, F1 score macro, F1 score micro, and weighted F1 score metrics. The experimental study conducted as a part of this work demonstrates the C1DCNN outperforms DNN and BD-LSTM-RNN for Opportunity, PAMAP2, Daphnet Gait, UCI-HAR, and SPHERE datasets with an f1-score 0.8912, 0.9444, 0.8876, 0.9637, and 0.8863 respectively.

Keywords: 1 Dimensional convolutional neural network (1D-CNN), Activation functions, Bi-directional long short-term memory (BD-LSTM), Deep learning (DL), Deep neural network (DNN), F1 score metrics, Human activity recognition.

1. Introduction

Human Activity Recognition (HAR) is one of the most challenging tasks and has attracted a lot of interest in the recent past. HAR is the key requirement in many important applications like health care [1], daily activity routines, providing care to senior citizens, etc. The data required to perform the tasks associated with HAR can be acquired using visual and non-visual sensors. Techniques employed for Computer Vision processing can be employed for HAR using the data collected from the visual sensors. Gathering data using visual sensors like cameras have their own challenges like a drastic change in light intensity, changes in camera angle and position, presence of obstacles and also issues of invasion of privacy, and these challenges limit their utility in real-life applications [2]. These limitations are overcome to a large extent by using non-visual sensors for HAR processes (NVHAR). NVHAR can be accomplished by using body-worn sensors, smart devices like smartwatches, mobile devices, object sensors, and environmental sensors.

In this perspective, sensors can be categorized into external sensors and wearable sensors. External sensors are placed in the immediate vicinity of the target while the wearable sensors are either carried or worn by the user. To recognize the activities, the user has to interact with those objects or has to be present in a certain location while performing the activity of interest. Then, the recorded sequence of sensor events is used to determine the activities that were performed. Recognition of simple activities like movements and postures can be accomplished by analysing the data collected by the sensors which are worn by the user. This approach cannot be adopted for the recognition of complex activities which involve various types of interaction with the environment. Hence, detecting these activities requires environmental data along with the data on the movement of various aspects of the body of the user. Smartphone-based HAR is also considered under the ambit of the wearable sensor-based approach.

Research in the recognition of physical activities with a wearable device approach has left some of the issues unaddressed. Some of such issues that are not adequately addressed are the decision of the users as to how the wearable device will be carried, significant differences in the movement patterns of persons based on their mood and also between different persons. In addition, it might be difficult to gather data from elderly and physically challenged persons. To address the aforementioned issues, most of the investigators maintained smartphones/wearable devices in immobile positions such as trouser pockets or on the waist of the user, or elsewhere on the human body. However, these wearables tend to hamper the freedom of movement of the user.

Overcoming these challenges and making the detection position and user-independent has attracted the attention of many researchers. Researchers have addressed this problem by extracting handcrafted features which are invariant or less influenced by the position of the smartphone and also by normalizing the data based on the position of the sensor. To achieve a reliable smartphone-based HAR, the performance of the existing approaches needs to be improved significantly.

The research on HAR is immensely benefited by conventional machine learning (ML) approaches like Random Forest [1, 2], KNN, SVM, MLP, Logistic Linear Regression [2], and Ensemble Neural Network [3] to infer human activities. In spite of their widespread use, Machine Learning (ML) - based approaches have major drawbacks, i.e., ML approaches completely rely on handcrafted features which

require a high degree of expertise in the domain to successfully build the HAR system. This limitation is overcome by adopting deep learning (DL) models for HAR.

The research on HAR is a significant element of Activity Recognition Systems (ARS) which help in Ambient Assisted Living [4]. These systems help in maintaining the quality of lifestyle of elders and the physically challenged. An enormous number of datasets of non-visual sensor-based HAR has been captured and shared by many investigators. As the success of HAR is strongly dependent on the environment it is essential to determine the relevant dataset for this evaluation process. This work provides a systematic methodology to evaluate deep learning algorithms on various HAR datasets. The contributions of this work are as follows:

- Proposed Customized 1D-CNN (C1DCNN) for recognition of human activities.
- Adopted DL models like DNN, and BD-LSTM for processing the HAR dataset.
- Fine-tuning of DL models to improve recognition accuracy.
- Exploring the ReLu based various activation functions on 1D-CNN.
- Evaluation of above DL models across HAR benchmark datasets like UCI-HAR, Opportunity, PAMP2, Daphnet Gait, and Sphere.
- Comparative performance analysis of DL models based on accuracy.

The rest of this article is arranged as follows: Section 2 Literature survey: It briefs about the state of art Deep Learning techniques employed to accomplish HAR and publicly available datasets to evaluate HAR models. Section 3 Methods and Material: It describes various deep learning models for HAR, and benchmark datasets used in this work. Section 4 Evaluation Metrics and Discussion: It describes performance evaluation metrics used in the present work to evaluate the performance of HAR models. Section 5 Experimental Results and Analysis: This section provides experimental results of the aforementioned deep learning models on various benchmark datasets.

2. Literature Survey

Upon the systematic study on the literature of the HAR, it has been observed that some reviews of the literature analyse very specific approaches from the technology point of view. Some reviews emphasize the systematic evaluation of various classification techniques' effectiveness for recognition of daily living activities from the datasets [5, 6]. Other reviews emphasize the systematic study on position-independent [7, 8] and user-independent [9] HAR. The further subsections of this section provide brief information about the aforementioned studies. Section 2.1 provides information concerning the traditional feature extraction and machine learning adopted for HAR, Section 2.2 reviews various deep learning approaches adopted for sensor-based HAR, and section 2.3 provides information about available benchmark datasets for sensor-based HAR.

2.1. Handcrafted features and machine learning models for sensor-based HAR

Damaševičius et al. [5] proposed the random projection-based approach for feature dimensionality reduction and low dimensionality features were trained using binary classifiers. In this work, researchers accomplished two HAR tasks namely, activity recognition and subject identification, with 95.52 percent

accuracy for within-person categorization and 94.75 percent accuracy for interpersonal identification respectively.

De Leonardis et al. [6] evaluated the five classification techniques in terms of HAR accuracy. For this evaluation, input signals were acquired using the MTx miniature magnetic and inertial measurement unit (MIMU) sensor manufactured by Xsens Technologies. A 5-s sliding window with no overlap was used to segment the input signal. Researchers extracted 21-time domain, 3 frequency domain, and 14 time-frequency domain features for each window. The evaluation showed that the result of the k-Nearest Neighbours (KNN) method had an accuracy of 97% this outperformed feedforward Neural Network (FNN) (95.8%), Support Vector Machine (SVM) (96.6%), Naive Bayes (NB) (96.5%), and Decision Tree (DT) (91%) for classification of 8 activities.

In smartphone-based HAR, smartphones were usually used in fixed positions like trouser pockets or on specific parts of the user's body. These constrain the user's behavior. To overcome this, Yang and Wang [7] proposed a method called PCAP (Parameters Adjustment Corresponding to smartphone Position). The calibration parameters of the accelerometer which is part of the smartphone are adjusted based on the position and orientation of the smartphone. To accomplish position-independent recognition, the SVM model was trained by using these features and obtained 91% accuracy for HAR. Saha et al. [8] evaluated shallow neural networks for position independent activity recognition of 7 different classes and achieved an accuracy of 75%. Table 1 provides the summary of HAR work carried out by various researchers using handcrafted features and machine learning classifiers.

Table 1. Summary of feature extraction and classifiers used for HAR.

Ref.	Input Dataset	Feature Extraction	Classifiers & Accuracy
[3]	UCAmI Cup Dataset	31 Features - one for each binary sensor	Ensemble NN- 80.39% KNN- 70.95% SVM - 76.54%
[9]	Constructed using Smartphone	21 features were extracted from magnitude acceleration sequences	Quadratic discriminant analysis- 95.4% kNN- 94.5%
[10]	Opportunity Dataset	Frequent temporal patterns (Apriori Principle)	SVM- 93% NB- 97% kNN- 91%
[11]	PAMAP Dataset	Time and frequency domain features and extracted mean and gradient from heart rate data	C4.5 - 98% Boosted C4.5 DT- 99% Bagging C4.5 DT- 98% NB - 93% kNN- 99%
[12]	UCI-HAR dataset	Enveloped power spectrum (EPS)	Linear Discriminant Analysis (LDA) - Multi-Class Support Vector Machine (MC-SVM)- 98.67%
[12]	DU-MD dataset	Enveloped power spectrum (EPS)	LDA - MC SVM- 100%
[13]	Constructed using TRIGNO EMG-acceleration	Wavelet energy spectrum features and ensemble-based filter feature selection	KNN– 80.4% SVM-81.7%
[14]	UCI-HAR	Time and frequency domain statistical features	MC-SVM- 96.81%

Siirtola and Röning [15] proposed a personalized Human Activity Recognition system using incremental learning approaches. In this work, data labelling was done using supervised, unsupervised, and semi-supervised techniques. An ensemble approach, Learn++ is adopted as a base classifier to train the labelled data. This study compares the results with 3 more base classifiers: linear discriminant analysis (LDA), classification and regression tree (CART), and quadratic discriminant analysis (QDA). These classifiers are evaluated using a publicly available dataset discussed by Siirtola and Röning [16]. Garcia-Gonzalez et al. [17] obtained 89.1% using the non-supervised personalization approach, 94.0% using the semi-supervised personalization approach, and 96.5% using the supervised personalization approach.

2.2. Deep learning (DL) models for sensor-based HAR

Due to the automatic feature extraction characteristic of DL, these algorithms are widely used in sensor-based HAR. In recent years, many researchers have utilized various deep learning algorithms to extract different types of information. The extraction of local translation-invariant features over a particular region or time can be accomplished by using a Convolutional Neural Network (CNN) [18]. Recurrent Neural Networks (RNN) is a neural network architecture that is appropriate for learning the temporal dynamics of sequential data. Murad and Pyun [19] established LSTM based 3 deep recursive neural network structures to recognize the activities by taking the time domain features of input sequences and achieved more accurate recognition. This network has the limitation of exploding and also displays an endangered gradient. To overcome this limitation, Ordóñez and Roggen [20] and Pienaar and Malekian [21 proposed the hybrid approach by combining the convolutional and LSTM recurrent neural network to perform the HAR using the data received from multimodal wearable sensors.

To accomplish wearable sensor-based HAR, Qian et al. [22] proposed a novel distributed neural network which can extract statistical features along with temporal and spatial information. Usually, DL algorithms involve plenty of operations, which makes them computationally expensive, hence it is not appropriate for real-time sensor-based HAR. Cheng et al. [23] proposed a computationally proficient CNN with conditionally parameterized convolution which maintains the balance of the trade-off between accuracy and high computational cost. Deep neural networks require high memory requirements when they are trained with global loss. This high memory requirement is reduced by training the networks with local loss. This concept is proposed by the investigators of [24] and accomplished for the usage of the HAR task using layer-wise CNN with local loss. To ameliorate the performance of HAR, features extracted from deep learning algorithms are combined with handcrafted features by considering regular dynamics of human conduct, and these features are trained using a maximum full a posterior (MFAP) scheme [25]. The sensor-based deep HAR system has been implemented on medium-range smartphone class hardware and extensive experiment have been carried out to analyse memory and execution time requirements [26]. Sensor-based HAR is employed to assist senior citizens and unwell people admitted to a hospital [27]. The zero-shot learning is applied to predict the "unobserved" activities which were not present in the training set [28] and evaluation is carried on the CASAS dataset and achieved high performance in recognizing hither to unseen (new) activities.

2.3. Benchmark dataset for sensor-based HAR

All advanced approaches of DL algorithms are evaluated using publicly available datasets and some of the researchers have created dataset in specific sensor-based environments. This section provides a few such datasets, considering the various characteristics such as number of activities, occupancy, context, number of participants, and type of sensors, with the subsequent abbreviations being used. O-Object Sensor, G-Gyroscope, M-Magnetometer, A- Accelerometer, E-Environmental Sensor. S- Single Occupancy, M-Multi Occupancy, NS- Not Specified (Table 2).

Table 2. Summary of publicly available datasets for sensor-based HAR.

Dataset	# Activities	Occupancy	Context	Type of Sensor	# Individual
van Kasteren et al. [29]	8	S	House	E	2
CASAS Multi resident [30]	8	M	Smart Apartment	E	2
CASAS Aruba [31]	11	M	Smart Workplace	E	1
CASAS Kyoto [32]	11	S	Smart Workplace	E	20
PAMAP2 [33]	18	M	Chest, Wrist and Ankle	A,G,M	9
Opportunity [34]	16	M	A room simulating studio flat	O,A	4
mHealth [35]	12	M	Chest, Wrist and Ankle	A,G	10
Daphnet Gait [36]	2	NS	Lab with emphasis on generating many freeze events.	A	10
WISDM [37]	6	M	Smart Phone and Smart Watch	A,G	29
SPHERE [38]	20	M	House	E, A	12

3. Methods and Material

This section consists of 4 sub-sections. Section 3.1 discusses the datasets used for the evaluation of DL algorithms. The architecture of DL algorithms is discussed in 3.2. The proposed work aims to assess the DL algorithms for sensor-based HAR using various benchmark datasets. Some of the DL work included in this work are DNN, CNN, and the Bi-directional LSTM RNN algorithm.

3.1. Datasets

In this study, we have employed publicly available datasets to build and evaluate DL algorithms. The datasets and their source links are listed in Table 3. Table 4 provides the information of the datasets employed in this work.

The objective of this study is to assess the DL algorithms for typical activities performed by humans in their daily life. Figure 1 describes the activities included in the above datasets. Except for the Daphnet Gait dataset, all datasets shown in Fig.1 consist of typical physical activities performed daily including static and dynamic activities. Daphnet Gait dataset includes only 3 activities namely, freeze, no freeze, and other. Labelling of activities in the SPHERE dataset has been done uniquely compared to the other datasets. Activities in the SPHERE dataset are prefixed with a_, p_, and t_, where 'a_' represents ambulatory activity (i.e., an activity involving ongoing movement), 'p_' represents static positions and 't_' represents the transition from one posture to another posture.

Table 3. Summary of publicly available datasets for sensor-based HAR.

Dataset	Source link
UCI HAR	https://archive.ics.uci.edu/ml/datasets/human+activity+recognition+using+smartphones#)
Opportunity Dataset	https://archive.ics.uci.edu/ml/datasets/OPPORTUNITY+Activity+Recognition#:~:text=Abstract%3A%20The%20OPPORTUNITY%20Dataset%20for,%2C%20feature%20extraction%2C%20etc
PAMAP2	https://archive.ics.uci.edu/ml/datasets/PAMAP2+Physical+Activity+Monitoring
Daphnet Gait	https://archive.ics.uci.edu/ml/datasets/Daphnet+Freezing+of+Gait
Sphere	https://github.com/IRC-SPHERE/sphere-challenge

Table 4. Summary of Datasets used in proposed work.

Dataset	# of Classes	# of Subjects	Sampling Rate	Training Window Length	# of Training Examples	# of Testing Examples
UCI-HAR	6	30	50Hz	128	11,988	2997
Opportunity	18	14	30 Hz	24	55,576	13,894
PAMAP2	18	9	100 Hz	25	255481	63871
Daphnet FOG	2	10	64 Hz	32	57,012	14,253
SPHERE	20	12	20 Hz	30		

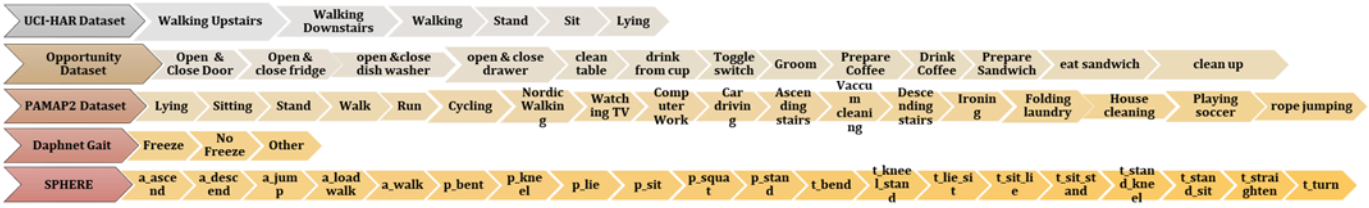

Fig. 1. Activities considered in the publicly available dataset

3.2. Deep learning approaches for HAR

3.2.1. Deep neural network

The deep feed-forward neural network for HAR has been implemented with four hidden layers accompanied by the output layer with softmax activation to classify the activities. The given input signal is segmented into equal width samples and each segmented data is considered as a feature. The input data is subjected to the sequence of non-linear transformations. Each hidden layer comprises an identical number of units and conforms to a linear transformation and a rectified linear

(ReLu) activation function [39]. The architecture of DNN employed in this work is shown in Fig. 2. The DNN network has been trained by varying the hyperparameters like learning rate, training epochs, and batch size. The network has been optimized by using Adam and Gradient Descent optimizers.

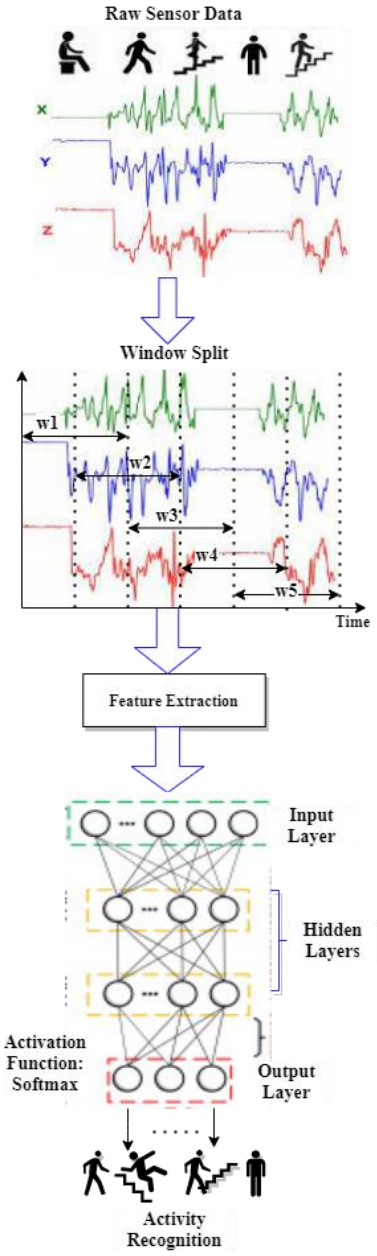

Fig. 2. The architecture of DNN for HAR.

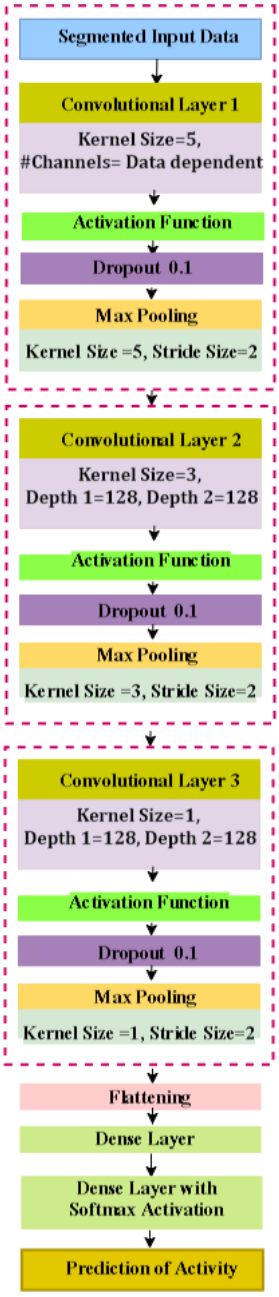

Fig. 3. The architecture of CNN for HAR.

3.2.2. Customized convolutional neural network (C1DCNN)

The CNN architecture mainly consists of three blocks namely, the Convolutional layer, Activation Layer, and Pooling layer. Figure 3 above depicts the architecture of the customized CNN designed for HAR. The functionality of each layer of the CNN is discussed in the subsequent part of this section.

The purpose of using the CNN algorithm for HAR is to obtain the translation-invariant features concerning the precise time of occurrence. Each convolutional layer extracts the features which are subjected to the Activation function followed by the dropout and Max Pooling layers. In this work, 1-dimensional sensor data is used as input hence one-dimensional kernel is employed to perform convolution for feature extraction. The convolution layer performs convolution on the input sequence using 'n' number of different kernels with a specific width 'w'. Figure 4 depicts steps involved in the feature extraction process. From Fig. 4, the input data sequence d_1, d_2,…..,d_n is segmented into "n" number of segments of equal length. Consider the convolutional kernel with kernel size 3 where w_1, w_2, and w_3 represent the weight coefficient of the kernel. Feature extraction is accomplished by performing the convolution on input data with respect to the kernel coefficient. The predefined operation will be performed on the data covered under the kernel coefficients. For example, from Fig. 4 the feature f_2 can be computed by taking the linear sum of the product of the weight coefficients and input data, for instance, $f_2 = w_1*d_1+w_2*d_2+w_3*d_3$.

The next building block of CNN is the activation function. Each convolutional layer in CNN performs a simple linear transformation of the input data. In absence of a nonlinear activation function after each layer, the entire network behaves as a simple linear transformation, which limits the accuracy of detection while performing complex tasks. To train the network for a complex task, a nonlinear activation function is used after each convolution layer. ReLu is one such nonlinear activation function with specific kernel width looking for introducing non-linearity into the convolution features by replacing the negative values with zero. The main drawback of the ReLu activation function is, it creates dead neurons those never get activated. To overcome this limitation, the C1DCNN employed Leaky ReLu, ELU, SeLu and sigmoid activation functions to introduce the non-linearity into the features extracted by the convolutions layers.

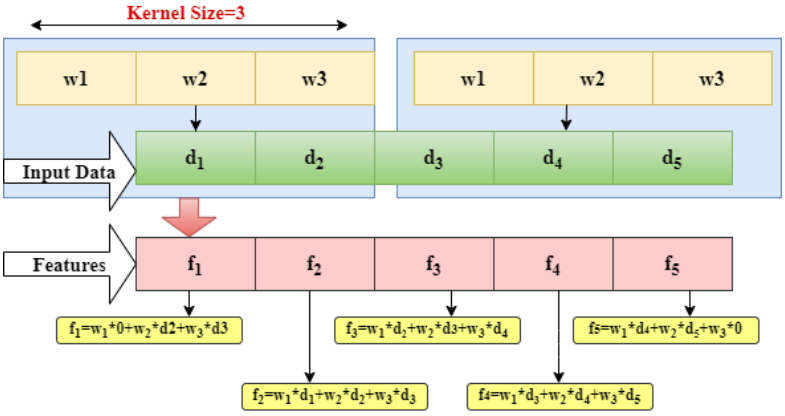

Fig. 4. Feature extraction in CNN.

The subsequent building block of the CNN is the pooling layer also called down-sampling. The Pooling layer is added after dropout in the CNN architecture as shown in Fig. 3. Pooling layers perform a reduction in the size of the activation maps. These layers are utilized after several steps of other layers (i.e., convolutional and non-linearity layers) to minimize the computational requirements progressively through the network as well as minimizing the likelihood of overfitting. The pooling layer has two hyperparameters: the size of the window and the stride. The size of the window and stride are defined well in advance. The pooling layer reduces the data within the window to a single value. After each iteration, the window is moved by the number of positions as defined by stride. This reduction is repeated at every position of the window until the whole activation volume is spatially reduced. Pooling performs the reduction by using two methods namely: max pooling and average pooling [40]. Max-pooling is performed with a specific kernel size, which determines the maximum value (max_value) within the region of the kernel of width (w), and the entire data of the window is represented max_value.

Similarly, the average pooling performs the reduction by reducing the data of the window to a single value by computing the average of the values of all the elements in the window. The subsequent fully connected neural networks effectively resemble a DNN and the architecture of the same is shown in Fig. 3.

3.2.3. Bi-directional long short-term memory (BD-LSTM)

In real life, human activity is continuous in nature. To predict the human dynamics of the human activities one should know the former and current states of the body. This can be accomplished by using baseline LSTM. Baseline LSTM depends solitarily on the information about the former state to predict the present status of the activity [39, 41]. Baseline LSTM cell will not be able to utilize the information of consequent state, this limitation is overcome by employing Bidirectional LSTM cells. The current work employs bidirectional LSTM because of its enhancement that the present output is related to both former and consequent state information. Bidirectional LSTM consists of two LSTM cells and both of these cells are responsible for the production of the output.

The data captured from the wearable and environment sensors are continuous in nature, i.e., not discrete. This data is segmented by using a window of width 'w' with 50% of overlapping. For interpretation, we define the input as $\{i_0, i_1, \ldots\ldots i_t, i_{t+1}, \ldots..\}$ the hidden layer as $\{h_0, h_1, \ldots\ldots h_t, h_{t+1}, \ldots..\}$ and output as $\{o_0, o_1, \ldots\ldots o_t, o_{t+1}, \ldots..\}$, where "t" represents the time instant. X represents the weight matrix from the input layer to the hidden, W represents the weight matrix from one hidden to the next hidden layer, and Y represents the weight matrix from the final hidden layer to the output layer. The forward and backward sequences in hidden layers are represented by $\vec{h}$ and $\overleftarrow{h}$ respectively. For the moment t (t=0,1,2….), the hidden layer and output layer are defined as given below:

$$\vec{h}_t = g(X_{\vec{h}} i_t + W_{\vec{h}} \vec{h}_{t-1} + b_{\vec{h}}) \quad (1)$$

$$\overleftarrow{h}_t = g(X_{\overleftarrow{h}} i_t + W_{\overleftarrow{h}} \overleftarrow{h}_{t-1} + b_{\overleftarrow{h}}) \quad (2)$$

$$o_t = g(Y_{\vec{h}} \vec{h}_t + Y_{\overleftarrow{h}} \overleftarrow{h}_t + b_o) \quad (3)$$

where g (.) represents the activation function and b represents the bias.

Equation (1) represents the forward sequence calculation from input to hidden or hidden to hidden and Eq. (2) denotes the backward sequence calculation from hidden to hidden or hidden to input. Equation (3) represents the calculation using both former and subsequent information. The results of the h_t computations are concatenated and then applied to the global max pooling to reduce the features. This will be followed by a fully connected neural network and finally a softmax function will perform the activity classification. The architecture of this network is depicted in Fig. 5.

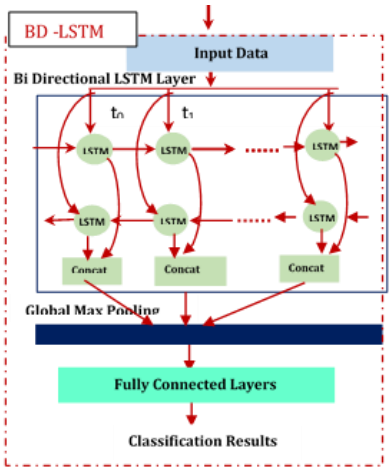

Fig. 5. The architecture of BD-LSTM.

4. Evaluation Metrics and Discussion

Five different benchmark datasets are employed to evaluate the deep learning models. Precision, Recall, and F1 measure metrics are used to evaluate the performance of these models. These metrics are calculated in 3 different ways namely micro, macro [42], and weighted. Precision is the measure of the ability of the classifier to classify the true positive samples as positive. Recall computes the number of correct class predictions in the dataset produced from all the positive classes. F1 score is the weighted average between precision and recall. The difference between micro and macro computing is that the former approach considers all the classes similarly thereby not discriminating between the different classes while the second one considers the size of each class also in the process of classification. The micro averaging approach just computes the averages of the metric score whereas macro averaging computes the metrics for each label and determines their unweighted mean. The formula of micro and macro precision is given in Eq. (4) and (5).

$$Precision_{micro} = \frac{TP_1+TP_2 \ldots\ldots+TP_n}{TP_1+TP_2 \ldots\ldots+TP_n+FP_1+FP_2 \ldots\ldots+FP_n} \quad (4)$$

$$Precision_{macro} = \frac{Precision_{\ 1}+Precison_2 \ldots\ldots+Precision_n}{n} \quad (5)$$

The equation of the micro and macro Recall is given in Eq. (6) and (7).

$$Recall_{micro} = \frac{TP_1 + TP_2 \ldots\ldots + TP_n}{TP_1 + TP_2 \ldots\ldots + TP_n + FN_1 + FN_2 \ldots\ldots + FN_n} \quad (6)$$

$$Recall_{macro} = \frac{Recall_{1} + Recall_{2} \ldots\ldots + Recall_{n}}{n} \quad (7)$$

The equation of the micro and macro F1 score is given in Eq. (8) and (9).

$$F_1Score_{micro} = 2 * \frac{Precision_{micro} * Recall_{micro}}{Precision_{micro} + Recall_{micro}} \quad (8)$$

$$F_1Score_{macro} = 2 * \frac{Precision_{macro} * Recall_{macro}}{Precision_{macro} + Recall_{macro}} \quad (9)$$

The weighted F-score is computed by considering the metrics for each label and finding their average, weighted by support (the number of true instances for each label). This alters macro computations to account for label imbalance; it can result in an F-score that is not between precision and recall.

5. Experimental Results and Analysis

5.1. Deep neural network

Assessment of the proposed model is accomplished by using the DNN model. To perform cross-validation, 80% of the dataset is utilized for training, and the remaining 20% of the dataset is utilized for testing. To determine appropriate activation functions for employed datasets, the experiment was conducted on various activation functions like ReLu, ELU, SeLu, Leaky Relu, and sigmoid with 200 epochs and a batch size of 64. The experimental results of this are depicted in Fig. 6.

From the analysis shown in Fig. 6 activity detection using DNN on the Opportunity dataset reached the highest training accuracy of 95.96 for the ELU activation function. The highest training accuracy for the PAMAP2 dataset is obtained using the ELU activation function. Similarly, the Daphnet Gait and UCI-HAR datasets obtained the highest training accuracy of 95.25, 97.25 with the ReLu activation function, and the SPHERE dataset obtained the highest training accuracy of 84.63 with ELU activation functions. Similarly, testing accuracy is evaluated on the same dataset for the above-mentioned activation functions. The experimental results are depicted in Fig. 7.

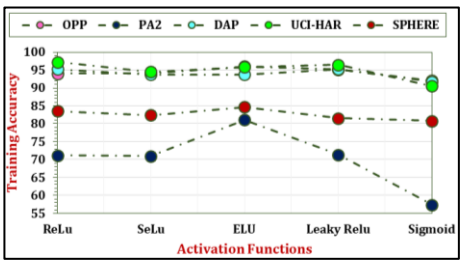

Fig. 6. Training Accuracy on various Activation Functions for DNN.

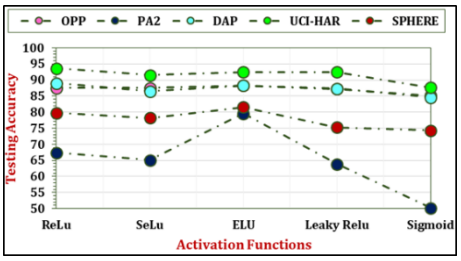

Fig.7. Testing Accuracy on various Activation Functions for DNN.

*OPP - Opportunity Dataset, *PA2 -PAMAP2 Dataset, *DAP- Daphnet Gait

Analysis of training and testing accuracy depicted in Figs. 6 and 7 show that OPP, PA2, and SPHERE datasets attained better training and testing accuracy using the ELU activation function; DAP and UCI-HAR datasets attained better training

and testing accuracy using the ReLu activation function. Hence, these datasets are evaluated for the corresponding activation function which gives better accuracy.

To select an appropriate optimizer in the layers of DNN, the experimentation is conducted on datasets by employing Adam and Gradient Descent optimizer with 200 epochs and a batch size of 64. The experimental results of this are shown in Figs. 8 and 9.

As per Figs. 8 and 9, the Adam optimizer outperforms the Gradient Descent optimizer. To select the appropriate number of hidden layers for deep neural networks, the experiment was conducted using the various number of hidden layers with Adam optimizer, 0.0001 Learning rate, and ELU activation function for Opportunity dataset, PAMAP2 dataset, and Sphere dataset; ReLu activation function for Daphnet gait and UCI-HAR dataset with 200 epochs and batch size of 64. Figures 10 and 11 illustrate the performance of the DNN for hidden layers 1 to 4. The training and testing accuracy of DNN is maximum for 2 hidden layers. However, it starts decreasing for a higher number of hidden layers. Hence, a neural network with two hidden layers is employed for further experimentation.

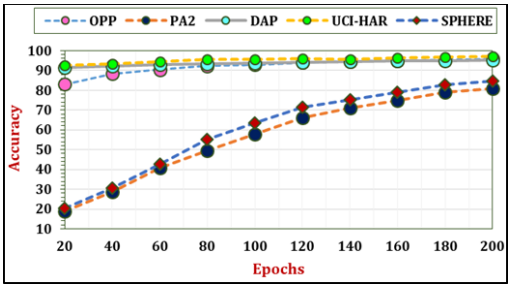

Fig. 8. Performance evaluation of DNN with Adam optimizer.

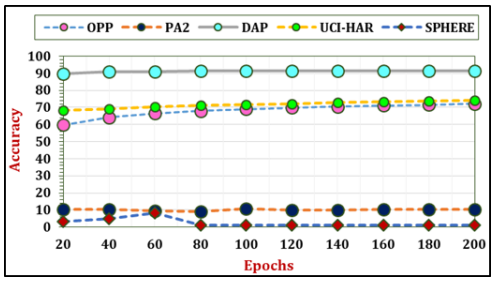

Fig. 9. Performance evaluation of DNN with gradient descent optimizer.

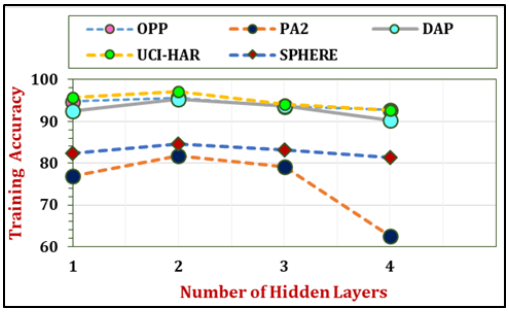

Fig. 10. Training Accuracy of DNN with number of hidden layers.

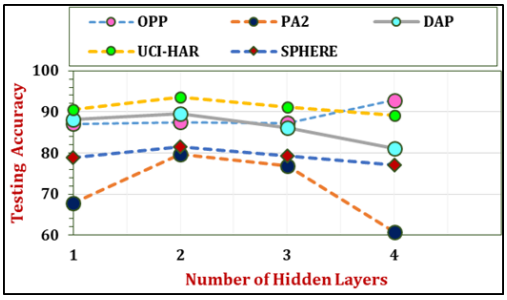

Fig. 11. Testing Accuracy of DNN with number of hidden layers.

5.2. Performance analysis of CNN for benchmark datasets

C1DCNN architecture is fine-tuned by choosing the appropriate optimizer and varying the parameters like the learning rate, and the number of epochs. For cross-validation, the dataset was divided into two parts in a ratio of 80:20. The training phase utilized 80% of the dataset and validation/ testing was performed on 20% of the dataset. To identify the suitable optimizer function, learning rate, and the

number of epochs for various datasets the experiment was conducted on various datasets with a batch size of 64.

All datasets were trained using CNN for 200 Epochs using Adam and Gradient Descent optimizer with a learning rate of 0.0001. The results of this experimentation are depicted in Figs. 12 and 13. As per Figs.12 and 13, the Adam optimizer provides higher accuracy.

Further, the network is fine-tuned by varying the learning rate with the step size of 0.0001. The Performance of CNN is evaluated for the learning rate 0.0001, 0.0002, 0.0003, 0.0004, and 0.0005 which is depicted in Table 5.

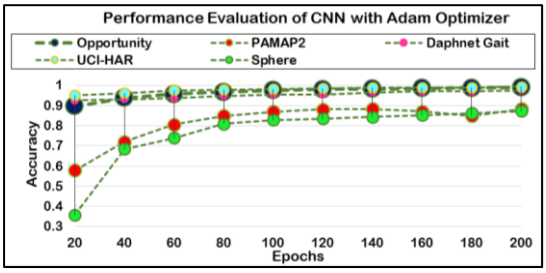

Fig. 12. Performance evaluation of CNN with Adam optimizer.

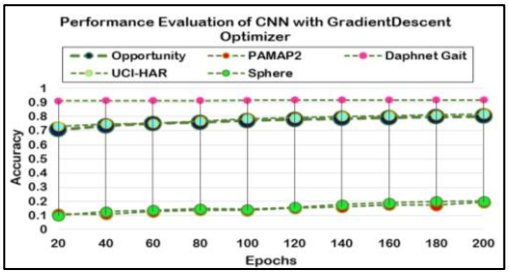

Fig. 13. Performance evaluation of CNN with gradient descent optimizer.

Table 5. Performance analysis of CNN with learning rate

UCI-HAR Dataset	**Learning Rate**	**0.0001**	**0.0002**	**0.0003**	**0.0004**	**0.0005**
	Epoch	182	150	136	129	95
	TR	99.3	98.9	97.5	98.2	97.8
	TS	93.5	92.8	90.7	93.1	90.9
OPPORTUNITY Dataset	**Learning Rate**	**0.0001**	**0.0002**	**0.0003**	**0.0004**	**0.0005**
	Epoch	198	105	125	110	100
	TR	99.11	98.89	98.73	97.9	97.5
	TS	92.45	89.85	91.48	91.2	90.35
PAMAP2 Dataset	**Learning Rate**	**0.0001**	**0.0002**	**0.0003**	**0.0004**	**0.0005**
	Epoch	129	85	59	59	49
	TR	89.6	89.3	84.5	82.8	83.3
	TS	94.3	94.1	91.8	91.4	82.3
Daphnet Gait Dataset	**Learning Rate**	**0.0001**	**0.0002**	**0.0003**	**0.0004**	**0.0005**
	Epoch	198	187	175	199	198
	TR	98.14	97.73	97.28	97.9	97.68
	TS	89.1	88.62	88.14	89.23	88.73
SPHERE Dataset	**Learning Rate**	**0.0001**	**0.0002**	**0.0003**	**0.0004**	**0.0005**
	Epoch	197	198	193	190	192
	TR	87.6	86.9	86.4	85.8	86.1
	TS	83.2	82.8	82.3	80.8	81.5

*TR- Training Accuracy TS-Testing Accuracy

The C1DCNN is finally trained with Adam optimizer with a learning rate of 0.0001, batch size 64 and with different activation functions like ReLu, SeLu, ELU, Leaky ReLU and sigmoid activation functions. The training and testing accuracy

of C1DCNN aforementioned activation functions is given in Table 6. As shown in Table 6, The Opportunity dataset achieved the highest 99.4% and 93.5% training and testing accuracy respectively by using SeLu activation function. For PAMAP2 dataset, the C1DCNN with ReLu activation function obtained the comparatively good accuracy of 89.6% and 94.3% training and testing accuracy respectively by using ReLu. Daphnet Gait dataset achieved the training and testing accuracy of 98.14% and 89.1% respectively using ReLu activation function. The C1DCNN with ELU activation function obtained the training accuracy 99.45% and 88.9% for UCI-HAR and SPHERE datasets respectively and the testing accuracy was 94.01% and 84.6% for UCI-HAR and SPHERE datasets.

Table 6. Training and testing accuracy on various functions for CNN.

Dataset	Training Accuracy					Testing Accuracy				
	ReLu	SeLu	ELU	Leaky Relu	Sigmoid	ReLu	SeLu	ELU	Leaky Relu	Sigmoid
OPP	99.1	99.4	98.7	99.2	89.2	92.4	93.5	91.28	91.9	90.3
PA2	89.6	87.3	88.6	87.6	82.1	94.1	93.2	93.5	93.5	91.2
DAP	98.1	97.2	97.5	97.5	90.28	89.1	85.1	85.23	84.45	82.3
UCI-HAR	99.3	97.2	99.4	98.3	92.6	93.5	92.8	94.01	93.12	89.6
SPHERE	87.6	83.4	88.9	82.6	80.4	83.2	79.6	84.6	78.4	75.3

5.3. Bidirectional LSTM recurrent neural network (BD-LSTM-RNN)

To accomplish HAR, the BDLSTM is employed with one forward and one backward RNN LSTM cell with 128 nodes in each layer. The learning rate considered for this work is 0.0001 and performance is analysed with Adam and gradient descent optimizers with epochs 500 and batch size of 64. Figure 14 depicts the training and testing accuracy for various benchmark datasets of HAR using Adam and Gradient descent optimizers and proves that Adam outperforms the other optimizers.

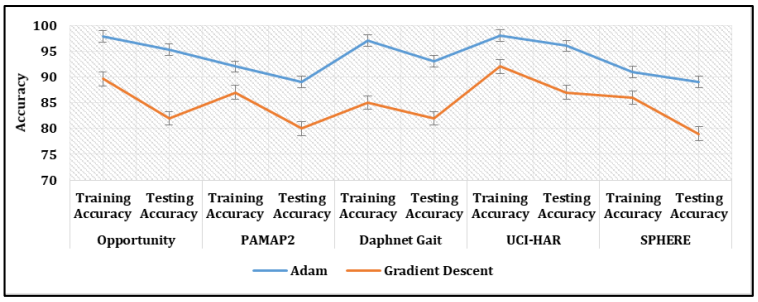

Fig. 14. Performance evaluation of BDLSTM with optimizer techniques.

5.4. Comparative analysis using evaluation metrics

In this section, Deep learning algorithms DNN, CNN, and BDLSTM are evaluated using evaluation metrics explained in section 4.

According to the experimental results shown in Table 7, the approach giving the best performance for the Opportunity dataset is C1DCNN with an F1-score of 0.891. BDLSTM is a considerably good approach for the Opportunity dataset with an F1score of 0.890. Similarly, C1DCNN outperforms on the PAMAP2, Daphnet Gait, UCI-HAR, and SPHERE datasets with f1-scores of 0.944, 0.887, 0.963, and 0.886 respectively.

Table 7. Comparative analysis using evaluation metrics.

Quality Measures	OPPORTUNITY Dataset			PAMAP2 Dataset			UCI- HAR Dataset		
	DNN	CNN	BD LSTM	DNN	CNN	BD LSTM	DNN	CNN	BDL STM
P_{micro}	0.88	0.884	0.883	0.812	0.944	0.887	0.886	0.963	0.894
P_{macro}	0.567	0.630	0.628	0.802	0.938	0.645	0.822	0.957	0.640
$P_{Weighted}$	0.88	0.906	0.899	0.817	0.948	0.916	0.824	0.967	0.914
R_{micro}	0.535	0.884	0.883	0.812	0.944	0.887	0.886	0.963	0.894
R_{macro}	0.880	0.700	0.699	0.785	0.936	0.725	0.853	0.955	0.712
$R_{Weighted}$	0.535	0.884	0.883	0.812	0.944	0.887	0.886	0.963	0.894
$F1score_{micro}$	0.880	0.884	0.883	0.812	0.944	0.887	0.886	0.963	0.894
$F1score_{macro}$	0.541	0.639	0.599	0.787	0.935	0.6486	0.812	0.954	0.658
$F1\ score_{Weighted}$	0.879	0.891	0.890	0.807	0.944	0.8967	0.862	0.963	0.901
	Daphnet Gait Dataset			SPHERE Dataset					
P_{micro}	0.861	0.897	0.728	0.832	0.878	0.815			
P_{macro}	0.668	0.789	0.718	0.816	0.872	0.603			
$P_{Weighted}$	0.841	0.885	0.732	0.821	0.882	0.851			
R_{micro}	0.861	0.897	0.728	0.832	0.878	0.815			
R_{macro}	0.617	0.691	0.707	0.798	0.870	0.652			
$R_{Weighted}$	0.861	0.897	0.728	0.832	0.878	0.815			
$F1score_{micro}$	0.861	0.897	0.728	0.832	0.878	0.815			
$F1score_{macro}$	0.635	0.726	0.715	0.799	0.869	0.623			
$F1\ score_{Weighted}$	0.849	0.887	0.736	0.812	0.886	0.826			

5.5. Results and Discussion

In this section, the performance of deep learning algorithms employed in this work for HAR is compared with other algorithms. Table 8 lists the accuracy of various algorithms used for HAR on various benchmark datasets.

It clearly shows that for the UCI HAR dataset the performance of the proposed algorithms CNN with Swish and ELU activation function outperforms the other algorithms studied in the literature review. Similarly, for the Opportunity dataset, proposed DNN, BD-LDTM-RNN, and C1DCNN with SeLU activation function outperform the performance of other algorithms with the accuracy of 95.96%, 97.8%, and 99.4% respectively.

For the PAMPA2 dataset, C1DCNN with a swish activation function outperforms other methods with an accuracy of 94.2%. For, the Daphnet Gait dataset, C1DCNN with a ReLu activation function outperforms other methods with an accuracy of 98.12%.

Table 8. Performance analysis of proposed methods and literature work.

Ref.	Dataset	Method	Accuracy
[10]	Opportunity	SVM	93%
[10]	Opportunity	Naïve Bayes	97%
[10]	Opportunity	KNN	91%
[18]	UCI	CNN + stat. features, intervals of size 50	94.35%
[18]	UCI	CNN + stat. features, intervals of size 128	97.63%
[19]	UCI	DRNN	96.50%
[19]	Opportunity	DRNN	92%
[19]	PAMPA2	DRNN	93%
[20]	Opportunity	DeepConvLSTM	93%
[22]	Opportunity	DDNN	86.1%
[22]	Opportunity	b-LSTM-S	92.7%
[22]	PAMPA2	DDNN	93.38%
[22]	Daphnet Gait	DDNN	92.5%
[22]	Daphnet Gait	b-LSTM-S	74.1%
[23]	PAMPA2	Conditionally Parametrized Convolutions	94.01%
[43]	UCI	PCA+SVM	91.82%
[44]	UCI	CNN	94.79%
[45]	UCI	Recurrent Neural Network	95.03%
[46]	Opportunity	AROMA	93.3%
[46]	Opportunity	DCNN	91.9%
[46]	Opportunity	DeepConvLSTM	91.2%
[47]	Opportunity	CNN-LSTM-ELM	91.8%
[48]	Opportunity	MLP-M	91.28%
[48]	Opportunity	CNN-M	90.88%
[48]	Opportunity	LSTM-M	92.30%
[48]	PAMPA2	MLP-M	82.47%
[48]	PAMPA2	CNN-M	93.74%
[48]	PAMPA2	LSTM-M	86.00%
[49]	PAMPA2	Dynamic Fusion-Ternary(2-bit)-Convolutional Network	91.40%
[50]	PAMPA2	LSTM + Continuous Temporal + Continuous Sensor Attention	89.96%
	Daphnet Gait		83.73%
Proposed	UCI	CNN (Swish Activation)	99.58%
Proposed	UCI	CNN(ELU)	99.45%
Proposed	Opportunity	DNN	95.96%
Proposed	Opportunity	BD-LSTM-RNN	97.8%
Proposed	Opportunity	CNN(SeLu)	99.4%
Proposed	PAMPA2	DNN	81.04%
Proposed	PAMPA2	BDLSTM	91.2%
Proposed	PAMPA2	CNN (Swish)	94.2%
Proposed	Daphnet Gait	DNN	95.25%
Proposed	Daphnet Gait	BDLSTM	91.2%
Proposed	Daphnet Gait	CNN	98.12%

6. Conclusion

This research work introduced the Customized 1D CNN architecture for HAR and also explored the state of the art of deep learning approaches to wearable sensor human activity recognition datasets. Comprehensive experiments were conducted on various benchmark datasets of HAR using customized CNN with different activation functions. In this experimental study, we determined that customized CNN with Swish, SELU, ELU, and Relu activation functions outperform the existing approaches in terms of accuracy for the aforementioned benchmark datasets. It is also seen that the BD LSTM approach is better than the deep neural networks-based configuration for all five datasets which are considered in this study.

Acknowledgments

The authors express their heartfelt thanks to Prof N.R Shetty, Advisor, Dr. H.C Nagaraj, Principal, and Mr. Rohit Punja, Administrator, Nitte Meenakshi Institute of Technology, affiliated to the Visvesvaraya Technological University (VTU), Belagavi, Karnataka for giving constant encouragement and support to carry out the research at NMIT.

References

1. Xu, L.; Yang, W.; Cao, Y.; and Li, Q. (2018). Human activity recognition based on random forests. 2017 13*th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery* (*ICNC-FSKD*), Guilin, China, 548-553.
2. Tran, P.H.; Nguyen, Q.T.; Tran, K.P.; and Heuchenne, C. (2020). Wearable sensor data based human activity recognition using deep learning: A new approach. *World Scientific Proceedings Series on Computer Engineering and Information Science Developments of Artificial Intelligence Technologies in Computation and Robotics*, *Cologne, Germany*, 581-588.
3. Irvine, N.; Nugent, C.; Zhang, S.; Wang, H.; and Ng, W.W.Y. (2020). Neural network ensembles for sensor-based human activity recognition within smart environments. *Sensors*, 20(1), 1-26.
4. De-La-Hoz-Franco, E.; Ariza-Colpas, P.; Quero, J.M.; and Espinilla, M. (2018). Sensor-based datasets for human activity recognition - A systematic review of literature. *IEEE Access*, 6, 59192-59210.
5. Damaševičius, R.; Vasiljevas, M.; Šalkevičius, J.; and Woźniak, M. (2016). Human activity recognition in AAL environments using random projections. *Computational and Mathematical Methods in Medicine*, Volume 2016 |Article ID 4073584, 1-17.
6. De Leonardis, G.; Rosati, S.; Balestra, G.; Agostini, V.; Panero, E.; Gastaldi, L.; and Knaflitz, M. (2018). Human activity recognition by wearable sensor : Comparison of different classifiers for real-time applications. 2018 *IEEE International Symposium on Medical Measurements and Applications* (*MeMeA*), Rome, Italy, 1-6.
7. Yang, R.; and Wang, B. (2016). PACP: A position-independent activity recognition method using smartphone sensors. *Information*, 7(4), 5-18.
8. Saha, S.S.; Rahman, S.; Haque, Z.R.R.; Hossain, T.; Inoue, S.; and Ahad, M.A.R. (2019). Position independent activity recognition using shallow neural architecture and empirical modeling. *UbiComp/ISWC '*19 *Adjunct: Adjunct Proceedings of the* 2019 *ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the* 2019 *ACM International Symposium on Wearable Computers*, 808-813.
9. Siirtola, P.; and Röning, J. (2012). User-independent human activity recognition using a mobile phone : offline recognition vs . real-time. *Proceedings of Nineth Distributed Computing and Artificial Intelligence*, 617-627.
10. Liu, Y.; Nie, L.; Liu, L.; and Rosenblum, D.S. (2016). From action to activity: Sensor-based activity recognition. *Neurocomputing*,. 181, 108-115.

11. Garcia-Gonzalez, D.; Rivero, D.; Fernandez-Blanco, E.; and Luaces, M.R. (2020). A public domain dataset for real-life human activity recognition using smartphone sensors. *Sensors*, 20(8), 1-14.
12. Bhuiyan, R.A.; Ahmed, N.; Amiruzzaman, M.; and Islam, M.R. (2020). A robust feature extraction model for human activity characterization using 3-axis accelerometer and gyroscope data. *Sensors*, 20(23), 1-17.
13. Tian, Y.; Zhang, J.; Wang, J.; Geng, Y.; and Wang, X. (2020). Robust human activity recognition using single accelerometer via wavelet energy spectrum features and ensemble feature selection. *Systems Science and Control Engineering*, 8(1), 83-96.
14. Ahmed, N.; Rafiq, J.I.; and Islam, M.R. (2020). Enhanced human activity recognition based on smartphone sensor data using hybrid feature selection model. *Sensors*, 20(1), 1-19.
15. Siirtola, P.; and Röning, J. (2019). Incremental learning to personalize human activity recognition models: The importance of human AI collaboration. *Sensors*, 19(23), 1-17.
16. Siirtola, P.; and Röning, J. (2012). Recognizing human activities user-independently on smartphones based on accelerometer data. *International Journal of Interactive Multimedia and Artificial Intelligence*, 1(5), 38-45.
17. Garcia-Gonzalez, D.; Rivero, D.; Fernandez-Blanco, E.; and Luaces, M.R. (2020). A public domain dataset for real-life human activity recognition using smartphone sensors. *Sensors*, 20(8), 24-26.
18. Ignatov, A. (2018). Real-time human activity recognition from accelerometer data using Convolutional Neural Networks. *Applied Soft Computing*, 62, 915-922.
19. Murad, A.; and Pyun, J.Y. (2017). Deep recurrent neural networks for human activity recognition. *Sensors*, 17(11), 1-17.
20. Ordóñez, F.J.; and Roggen, D. (2016). Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. *Sensors*, 16(1), 1-25.
21. Pienaar, S.W.; and Malekian, R. (2019). Human activity recognition using LSTM-RNN deep neural network architecture. *Proceedings of* 2019 *IEEE* 2*nd Wireless Africa Conference* (*WAC*), Pretoria, South Africa, 1-5.
22. Qian, H.; Pan, S.J.; Da, B.; and Miao, C. (2019). A novel distribution-embedded neural network for sensor-based activity recognition. *Proceedings of Twenty-Eighth International Joint Conference on Artificial Intelligence* (*IJCAI*-19), 5614-5620.
23. Cheng, X.; Zhang, L.; Tang, Y.; Liu, Y.; Wu, H.; and He, J. (2022). Real-time human activity recognition using conditionally parametrized convolutions on mobile and wearable devices. *IEEE Sensors Journal*, 22(6), 5889-5901.
24. Teng, Q.; Wang, K.; Zhang, L.; and He, J. (2020). The layer-wise training convolutional neural networks using local loss for sensor-based human activity recognition. *IEEE Sensors Journal*, 20(13), 7265-7274.
25. Chen, Z.; Xiang, S.; Ding, J.; and Li, X. (2020). Smartphone sensor-based human activity recognition using feature fusion and maximum full a posteriori. *IEEE Transactions on Instrumentation and Measurement*, 69(7), 3992-4001.

26. Zebin, T.; Scully, P.J.; Peek, N.; Casson, A.J.; and Ozanyan, K.B. (2019). Design and Implementation of a Convolutional Neural Network on an Edge Computing Smartphone for Human Activity Recognition. *IEEE Access*, 7, 133509-133520.
27. Schrader, L.; Toro, A.V.; Konietzny, S.; Rüping, S.; Schäpers, B.; Steinböck, M.; Krewer, C.; Müller, F.; Güttler, J. and Bock, T. (2020). Advanced sensing and human activity recognition in early intervention and rehabilitation of elderly people. *Journal of Population Ageing*, 13, 139-165.
28. Al Machot, F.; Elkobaisi, M.R.; and Kyamakya, K. (2020). Zero-shot human activity recognition using non-visual sensors. *Sensors*, 20(3), 1-17.
29. van Kasteren, T.L.M.; Englebienne, G.; and Kröse, B.J.A. (2010). Activity recognition using semi-Markov models on real world smart home datasets. *Journal of Ambient Intelligence and Smart Environments*, 2(3), 311-325.
30. Singla, G.; Cook, D.J.; and Schmitter-Edgecombe, M. (2010). Recognizing independent and joint activities among multiple residents in smart environments. *Journal of Ambient Intelligence and Humanized Computing*, 1(1), 57-63.
31. Cook, D.J. (2012). Learning setting-generalized activity models for smart spaces. *IEEE Intelligent Systems*, 27(1), 32-38.
32. Cook, D.J.; Crandall, A.S.; Thomas, B.L.; and Krishnan, N.C. (2013). CASAS: A smart home in a box. *Computer*, 46(7), 62-69.
33. Reiss, A.; and Stricker, D. (2012). Introducing a new benchmarked dataset for activity monitoring. *Proceedings of* 2012 16*th International Symposium on Wearable Computers*, Newcastle, UK, 108-109.
34. Chavarriaga, R.; Sagha, H.; Calatroni, A.; Digumarti, S.T.; Tröster, G.; Millán, J.D.R.; and Roggen, D. (2013). The opportunity challenge: A benchmark database for on-body sensor-based activity recognition. *Pattern Recognition Letters*, 34(15), 2033-2042.
35. Banos, O.; Garcia, R.; Holgado-Terriza, J.A.; Damas, M.; Pomares, H.; Rojas, I.; Saez, A.; and Villalonga, C. (2014). mHealthDroid: A novel framework for agile development of mobile health applications. *Lecture Notes in Computer Science* (*Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics*), 8868, 91-98.
36. Bächlin, M.; Plotnik, M.; Roggen, D.; Maidan, I.; Hausdorff, J.M.; Giladi, N.; et al. (2010). Wearable assistant for Parkinsons disease patients with the freezing of gait symptom. *IEEE Transactions on Information Technology in Biomedicine*, 14(2), 436-446.
37. Weiss, G.M.; Yoneda, K.; and Hayajneh, T. (2019). Smartphone and smartwatch-based biometrics using activities of daily living. *IEEE Access*, 7, 133190-133202.
38. Twomey, N.; Diethe, T.; Kull, M.; Song, H.; Camplani, M.; Hannuna, S.; Fafoutis, X.; Zhu, N.; Woznowski, P.; Flach, P.; and Craddock, I. (2016). The SPHERE challenge: activity recognition with multimodal sensor data, *arXiv preprint arXiv:*1603.00797, 1-14.
39. Hammerla, N.Y.; Halloran, S.; and Plötz, T. (2016). Deep, convolutional, and recurrent models for human activity recognition using wearables. *Proceedings*

of Twenty-Fifth International Joint Conference on Artificial Intelligence, New York, USA, 1533-1540.

40. Suárez-Paniagua, V.; and Segura-Bedmar, I. (2018). Evaluation of pooling operations in convolutional architectures for drug-drug interaction extraction. *BMC Bioinformatics*, 19(209), 39-47.
41. Zhao, Y.; Yang, R.; Chevalier, G.; Xu, X.; and Zhang, Z. (2018). Deep residual bidir-lstm for human activity recognition using wearable sensors. *Mathematical Problems in Engineering*, Volume 2018 |Article ID 7316954, 1-14.
42. Ponce, H.; Martínez-Villaseñor, M.D.L.; and Miralles-Pechuán, L. (2016). A novel wearable sensor-based human activity recognition approach using artificial hydrocarbon networks. *Sensors*, 16(7), 1-28.
43. Shi, D.; Li, Y.; and Ding, B. (2015). Unsupervised feature learning for human activity recognition. *Guofang Keji Daxue Xuebao*/*Journal of National University of Defense Technology*, 37(5), 128-134.
44. Ronao, C.A.; and Cho, S.B. (2016). Human activity recognition with smartphone sensors using deep learning neural networks. *Expert Systems with Applications*, 59, 235-244.
45. Kolosnjaji, B.; and Eckert, C. (2015). Neural network-based user-independent physical activity recognition for mobile devices. *Lecture Notes in Computer Science* (*Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics*), 9375, 378-386.
46. Peng, L.; Chen, L.; Ye, Z.; and Zhang, Y. (2018). AROMA : A deep multi-task learning based simple and complex human activity recognition method using wearable sensors. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2(2), 1-16.
47. Sun, J.; Fu, Y.; Li, S.; He, J.; Xu, C.; and Tan, L. (2018). Sequential human activity recognition based on deep convolutional network and extreme learning machine using wearable sensors. *Journal of Sensors*, Volume 2018 |Article ID 8580959, 1-19.
48. Lv, T.; Wang, X.; Jin, L.; Xiao, Y.; and Song, M. (2020). Margin-based deep learning networks for human activity recognition. *Sensors*, 20(7), 1871-1889.
49. Yang, Z.; Raymond, O.I.; Zhang, C.; Wan, Y.; and Long, J. (2018). DFTerNet: towards 2-bit Dynamic fusion networks for accurate human activity recognition. *IEEE Access*, 6, 56750-56764.
50. Zeng, M.; Gao, H.; Yu, T.; Mengshoel, O.J.; Langseth, H.; Lane, I.; and Liu, X. (2018). Understanding and improving recurrent networks for human activity recognition by continuous attention. *Proceedings of ACM International Symposium on Wearable Computers*, 56-63.