WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

DIVIDE AND CONQURER BASED 1-D CNN HUMAN ACTIVITY RECOGONIZITION USING DATA SHARPENING

Dr. J.B.Jona[1], Mr.S.A.Gunasekaran.[2], Hariramakrishnan S[3], Venkateshan D[3]

[1]Associate Professor Dept. Of Computer Applications, COIMBATORE INSTITUTE OF TECHNOLOGY
[2]Assistant Professor, Dept. Of Computer Applications, COIMBATORE INSTITUTE OF TECHNOLOGY
[3]MSc Decision and Computing Sciences, CIT
Email ID: jona@cit.edu.in; agunasekaran@cit.edu.in; hramakrishnan02@gmail.com; arunvenkatesh910@gmail.com

ABSTRACT

The recent advancements of Artificial Intelligence (AI) have make the human being more inclined towards novel research aims in recognizing the objects, learning the environment, time series analysis and predicting the forthcoming sequences. Nowadays there is a growing interest of AI researchers towards Recurrent Neural Networks (RNN) which compromises massive applications in the fields of speech recognition, language modeling, video processing and also time series analysis. Human Activity Recognition (HAR) is one of the challenging problems which seeks answers in this wonderful AI field. It can be mainly used for eldercare and childcare as an assistive technology combined with technologies like Internet of Things (IoT). Also, HAR occupies wide range of applications in real life ranging from healthcare to personal fitness, gaming, military applications, security fields etc. HAR can be done with sensors, images, smart phones or videos where capturing behaviours by using sensors such as accelerometers and gyroscopes has become more famous with the development of Human Computer Interaction (HCI) technology. This paper presents an approach to predict human activities developed using CNN and Long Short-Term Memory (LSTM) on the basis of the WISDM dataset.

Keywords *- Human Activity Recognition, Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM)*

1. INTRODUCTION

Human Activity Recognition is the process of identifying, analysing and interpreting what kind of actions and goals one or more agents or persons will be performing. The decisions will be taken based on their previous actions performed with their behaviour. A typical human may perform the major activities such as walking, running, sitting, standing, laying, walking-upstairs, walking-downstairs etc. in his or her day today routine. Identifying and analysing various human activities would be bringing out some smart solutions associated with childcare and eldercare fields if HAR could be integrated with IoT technologies and means. For example, suppose a situation where a child is kept at a day care centre, parents have been gone for work and they need to check what their child is actually doing right now or are they safe at this moment, this HAR could be used as a measure to predict

Human behaviour. Also, in case of elderly people monitoring guardians or caretakers could use this technology for making better safe environment for them by avoiding some actions elders tend to do. Thus, ranging from personal fitness. Gaming, security fields, health care industry and even more, HAR could bring attractive solutions for real life human problems.

With the recent emerging trend of Deep Learning, CNN and RNN architectures have become more predominant and to date the application of Deep Learning models to train time series of inertial sensor data is still under researcher's exploration. Deep Learning models such as CNN and RNN focus on a data-driven approach to learn discriminative features from raw sensor data to sequential information. Human activities normally are measured with sensors either be external or wearable such as accelerometers and gyroscopes. Accelerometer data measures people's speed of doing things and gyroscope data measures the angular velocity of the actions. Then since these sensors provide a large dataset development of proper automatized systems to process and analyse the entire dataset will be an important task. In this context, HAR systems will have an important task to prevent the data analysis issues associated with the system. From this large collected raw data, a feature vector will be extracted and at the end learning algorithms will be used to generate an activity recognition model based on the feature vector. Thus, selection of a well-trained, efficient model is essential to grasp the maximum accuracy with the recognition process.

The rest of this paper is organized as follows. Section 2 gives an overview of the dataset used, LSTM architecture, CNN LSTM architecture and the HAR paradigm. Section 3 gives our implications and methodology used to implement the system. Experiment results are shown under the section 4. Finally, paper concludes giving the conclusion cited and expecting future works associated with the learnings and results of the overall research.

2. LITERATURE REVIEW

A. WISDM Dataset:

The dataset used in the experiment is the standard WISDM dataset, which is also known as Smartphone and smartwatch activity, and Biometrics dataset. It contains accelerometer and gyroscope time series sensor data collected from a smartphone and smartwatch as 51 test subjects performing 18 activities for 3 minutes each with a 20Hz sampling rate. 36 users have been participated in the experiment. This is available and can be downloaded from the UCI machine-learning repository. The size of the dataset is 1,098,207 which contains data relate to 6 attributes as walking, jogging, upstairs, downstairs, sitting and standing. The columns are as user, activity, timestamp, acceleration, y-acceleration and z-acceleration. Originally this is an unbalanced dataset where walking contains 38.6% of data, Jogging contains 31.2% of data, upstairs include 11.2% data and Downstairs, Sitting and Standing contains 9.1%, 5.5% 4.4% of the data respectively.

B. CNN Architecture :

CNN architecture has been inspired by the basic structure and the functionality of the visual cortex of the human brain. A neuron in a layer will only be connected to a small region of the layer before it, instead of all the neurons in a fully connected layer. CNN are also known as Convent. CNN are not fully connected. The design of the conventional CNN includes the input layer, output layer and multiple hidden layers which can be the convolutional layer, relu layer, pooling layer and finally the fully connected layer. The final convolution often involves Backpropagation in order to efficiently converge the estimation error, and accurately weight the end result. The objective of the convolution operation would be to extract the high-level features which endeavour to provide more essential and delicate connection between the classification input and output.

C. LSTM Architecture:

LSTM is the type of RNN that can learn very long-term dependencies that are able to learn and remember over long sequences of input data. Thus, LSTM are commonly used for time series analysis problems. LSTM doesn't use activation functions within its recurrent components. There stored values are not modified. LSTM doesn't also have the vanishing gradient problem during training.

Usually LSTM are implemented in 'blocks' or cell and blocks have 3 or 4 gates such as input gate, output gate or forget gate *etc.* (Fig. 1). LSTM uses the concept of gating to deal with the vanishing gradient problem. The cell is capable of remembering values over different time intervals.

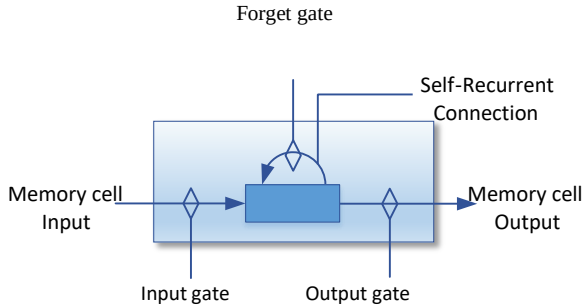


Fig. 1. Illustration of an LSTM memory cell

Each of these cells are considered as an artificial neuron with an activation function of a weighted sum of the current data x_t, a hidden state h_{t-1} from the previous time step, and any bias b (Fig. 2).

The benefit of using LSTMs for sequence classification is that they can learn from the raw time series data directly, and in turn do not require domain expertise to manually engineer input features. The model can support multiple parallel sequences of input data, such as each axis of the accelerometer and gyroscope data. The model learns to extract features from sequences of observations and how to map the internal features to different activity types.

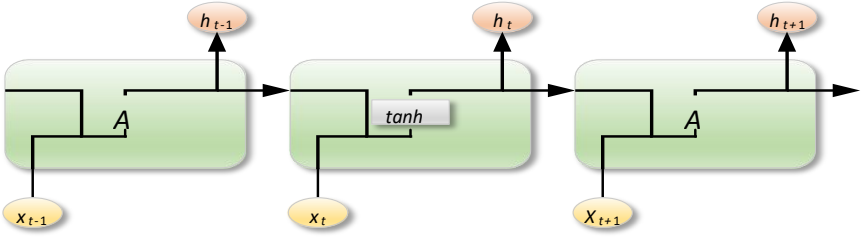


Fig. 2. The repeating module in a standard RNN which has a single layer

D. Human Activity Recognition with CNN and LSTM:

HAR has been extensively a wide research area in the past decade. HAR on smart phone data and various sensor data has been evolving rapidly around many multiple applications which benefit to human mankind. Advances in the area of mobile sensing enable users to quantify their sleep and exercise patters, monitor personal commute behaviours, track their emotional state or even any kind of human activity a person is involving in. Statistical learning methods have also been used to tackle with the activity recognition problems. Some of them are Naïve Bayes, K-Nearest Neighbour (KNN) which has been used to recognize seven motions such as walking, running and jumping etc. However, they have required expert knowledge to design the features and systems have become more heuristic.

Literature proves many successful implementations on HAR using CNN and LSTM models. In researchers describe a predictive model which use the UCI-HAR dataset for CNN layers for feature extraction on input data combined with LSTM to support sequence prediction. Also, they present an approach which uses convolutions directly as part of reading input into the LSTM units itself. CNN LSTM model have achieved an overall accuracy of 90.6% while Convolutional LSTM presented an accuracy of about 90%. Presents a methodology which uses CNN LSTM where the output of convolution is set as the input to LSTM. Convolution has been done on different time windows and via a "Time Distributed layer" that uses for time series analysis. Some probabilistic models for behaviour prediction has also been developed applying LSTM network for behaviour modelling.

In researchers have implemented and improved a stacked LSTM architecture for the feature-free classification of activities using both accelerometer and gyroscope signals as the raw data input. There an approach using CNN and LSTM has been carried out for the human activity recognition from inertial sensor time series using batch normalized deep LSTM recurrent networks is presented. LSTM model is a multi-stacked architecture LSTM network for multi class HAR classification.

In literature, the use of basic classification techniques for HAR is also can be seen. Firstly, logic based classification algorithms such as decision trees have shown an accuracy of 98.7% in applications. Perceptron based algorithms show an accuracy of 89%. Alternatively, statistical algorithms such as Naïve Bayes and Bayesian networks have shown an accuracy of 98% and 90.57% respectively. K-Nearest neighbour have had an average accuracy of 99.25% and 90.61%. Lastly SVM show an accuracy of 97.5%.

A HAR system also faces many challenges, such as large variability of a given action, similarity between classes, time consumption, and the high proportion of null values. All of these challenges have led researchers to develop representation methods of systematic features and efficient recognition methods to effectively solve these problems. In researchers proposed deep convolutional network with utilization of CNN and LSTM. This paper took advantage of LSTM to solve sequential human activity recognition problem and achieved a good precision. But the complex network framework suffered from low efficiency and can hardly meet real-time requirements in practice applications.

3. METHODOLOGY

With the literature review been conducted, it was revealed that the Deep Learning Models have been widely used resulting better scales of accuracy and to serve the Human Activity Recognition process. In addition, the neural network's robust nature, generalizable capability and scalability has inspired many researchers to apply deep learning in their machine learning models.

In our approach, the implementations were carried out under four main areas. First, developed a basic CNN model for the activity recognition. Second, developed a LSTM network model for our dataset. As the third step, developed a CNN and LSTM hybrid model for the classification and the prediction of the six activities. Finally, proposed a ConvLSTM model which is a further extension of the CNN LSTM model to perform the convolutions of the CNN as part of the LSTM. Prior to developing the models and carried a comprehensive detailed exploratory data analysis of the WISDM dataset.

A. Exploratory Data Analysis of Dataset

Exploratory data analysis (EDA) is the approach of analysing datasets to summarize the main characteristics presented in the data. Abbreviated, it is seeing what our data can tell us with discovering patterns with data. EDA is the backbone of any data science project which helps to understand the distribution of data which will be needed for better classification and prediction.

First, WISDM dataset was loaded in to Jupiter notebook environment. Here, several python open source libraries have been employed in the EDA analysis, including Pandas and Sklearn with various data processing functions. With the help of the Pandas library, records with missing values were removed. Figure 3 illustrates the count distribution of recorded activities per person after the missing-value removal.

Next, the count of each activity with respect to the six different activities was plotted as shown in the following pie chart (Fig. 4). This pie chart clearly shows the unbalanced nature presented in the raw WISDM dataset which would potentially cause for irregularities during the data classification and the prediction process.

A balanced dataset would always be better in a perfect classification and prediction process because assuring each of the representing classes would hold the same probability to occur without any biasness. Thus, WISDM dataset was balanced by selecting the same amount of data rows for each of the 6 activities which is graphically represented as the next pie chart (Fig. 5).

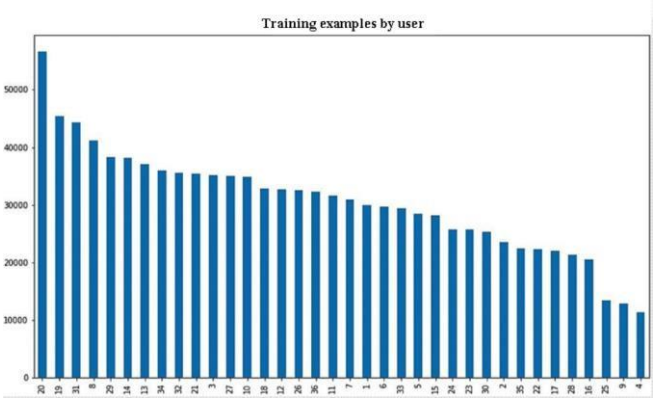


Fig. 3. Count of activities per person.

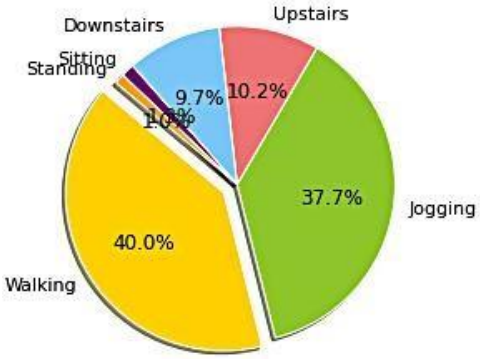


Fig. 4. Pie chart of record distribution by activity type with unbalanced data

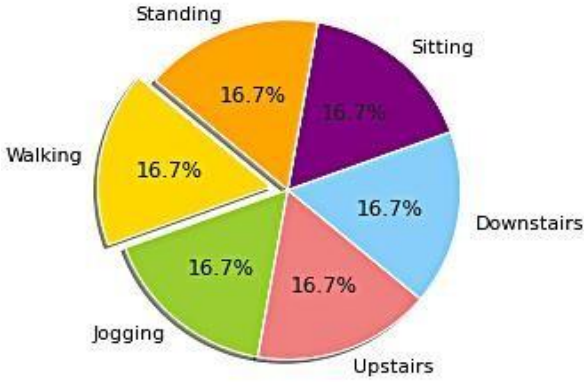
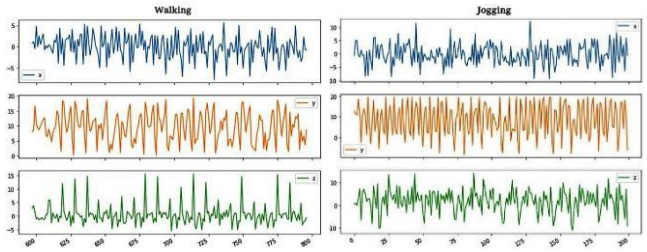


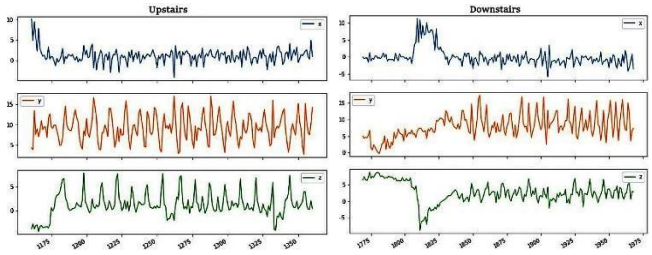
Fig. 5. Pie chart of record distribution by activity type with balanced data

Next, in order to identify the nature of the signals relating to x, y and z axes, the accelerometer data was plotted with respect to the 6 activities separately for their first 200 entries as shown in the following plots. With a clear visual inspection of the following plots, the differences in each axis of the signals across each activity can be identified well. It is seen that the nonstationary activities like walking, jogging, upstairs and downstairs have multiple variations with the signals while stationary activities, like sitting and standing, compasses only quite small amount of variations in their accelerometer signals, as those shown in Fig. 6.

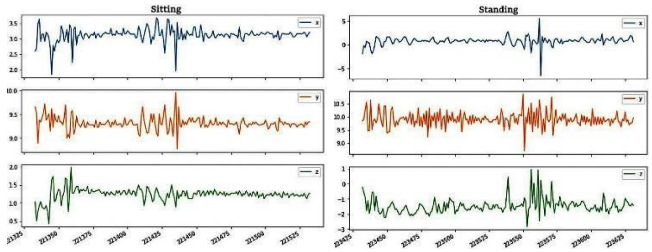
The activity column which is a categorical variable in the dataset was then converted in to the numerical format. For this purpose, the Label Encoder function from the Sklearn library was used for pre-processing. In the process of feature scaling, all the features were scaled to be within the same range, which would guarantee the value manipulations of every features equivalent and reweight naturally the prediction model by real dependency of the corresponding relevance of the features. Here, the Sklearn's StandardScaler function, which scale each feature by its maximum absolute value, was used for the scaling.



(a) Walking **(b) Jogging**



(c) Upstairs **(d) Downstairs**



(e) Sitting **(f) Standing**

Fig. 6. Variation nature of the signals for the six activities

Finally, the data will need to be prepared in a format required by the designate models. For this purpose, fixed sized frame segments were created from the raw signals. The procedure would generate indexes as specified by a fixed size of steps moving over a thread of signal. The fixed step-size was parameterized with 20 for this study. The frame size used is 80 (step-size x 4), which equals to 4 seconds of data, *i.e.*, elementary sample are created in 4 seconds per segment. The label (activity) for each segment is selected by the most frequent class label or generally the mode presented in that window accordingly. The resulted dataset in the desired format is then split with an 8:2 ratio for training and testing datasets, respectively.

B. CNN Architecture:

The CNN model was defined as having two CNN hidden layers. Each of them are followed by two dropout layers of 0.5 in order to reduce overfitting of the model to the training data. Then a dense fully connected layer is used to interpret the features extracted by the CNN hidden layers. Finally, a dense layer with the softmax activation function was added as the final layer to make predictions (Table I).

The sparse categorical cross entropy loss function will be used as the loss function and the efficient Adam version of stochastic gradient descent was used to optimize the network with a learning rate of 0.001. CNN model was trained for 50 epochs and a batch size of 64 samples were used. After the model is fit, it was evaluated on the test dataset and the accuracy of the CNN model was obtained.

TABLE I. THE DIMENSIONAL SRUCTURE OF THE ADOPTED CNN MODEL

Layer	Output Shape	Param #
Conv2D	None, 79, 2, 16	80
Dropout	None, 79, 2, 16	0
Conv2D	None, 78, 1, 32	2,080
Dropout	None, 78, 1, 32	0
Flatten	None, 2496	0
Dense	None, 64	159,808
Dropout	None, 64	0
Dense	None, 6	390
Total params: 162, 358 Trainable params: 162, 358 Non-trainable params: 0		

C. LSTM Architecture

The LSTM model was defined as having a single LSTM hidden layer. A dropout layer valuing 0.5 follows this. Then a dense fully connected layer is used to interpret the features extracted by the single LSTM hidden layer. Finally, a dense layer was added as the final layer to make predictions (Table

II).

For the purpose of compiling and training the LSTM model, the same values for the loss function, optimizer, batch size and the number of epochs, which used, in compiling and training the CNN model were used. After the model is fit, it was evaluated on the test dataset and the accuracy was obtained.

TABLE II. TABLE II. THE DIMENSIONAL SRUCTURE OF THE ADOPTED LSTM MODEL.

Layer	Output Shape	PARAM #
LSTM	None, 100	41600
Dropout	None, 100	0
Dense	None, 100	10100
Dense	None, 6	606
Total prams: 52,306 Trainable prams: 52,306 Non-trainable prams: 0		

4. EXPERIMENTAL RESULTS

A. Results from CNN and LSTM Models:

The implementation was realized under a Jupiter notebook environment of Google Colaboratory® by Python programming language. With the four model architectures described in the previous section, all the four models were compiled together with the sparse categorical cross entropy loss function and the Adam optimizer with a learning rate of 0.001. All the NN models was fitted for the training data and test data with a batch size of 64 and run for 50 epochs. The training accuracy was then plotted together with the validation accuracy varying the iterations for performance evaluation related to the two models (Fig. 7 and Fig. 8).

With respect to the CNN model, a training accuracy of 99.53% was achieved while the validation accuracy of 93.46% was simultaneously achieved as that shown in Fig. 7.

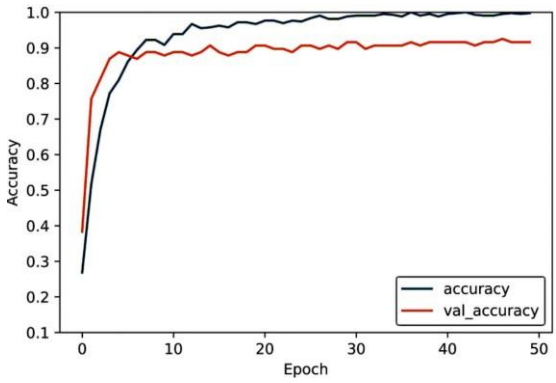


Fig. 7. Training and validation accuracies with the CNN model.

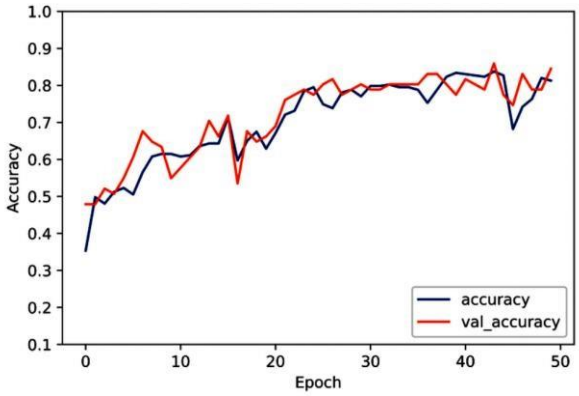


Fig. 8. Training and validation accuracies with the LSTM model.

Accompanying with the training and validation accuracies, the training and validation losses calculated during the procedure were also charted in a graph varying the number of iterations for all the four models. It is seen that both the training and validation losses are gradually decreasing with the iterations to converge to the approximation within respective precise ranges in both the models. The relative lower validation loss resulted in the two models guarantees that no overfitting happened to the converged models. Figures 9 and 10 show the training and validation loss resulted from the CNN model and LSTM model, respectively.

In addition to the accuracies, confusion matrices, which helps to graphically check the true label and the predicted label more comparatively, has also been constructed (Figs. 11 and 12). The confusion matrix contains information about the actual and predicted classifications done by a classification system and the performance of such systems is commonly evaluated using the data in the matrix. By checking on the test samples, the confusion matrices were charted as follows for the two models.

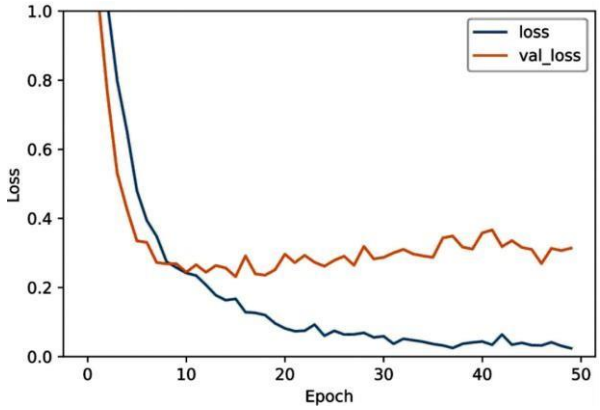


Fig. 9. Losses calculated during the iterative procedure for both training and validation with the CNN model.

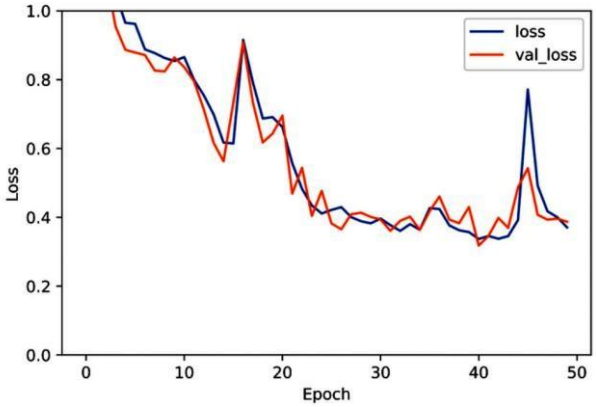


Fig. 10. Losses calculated during the iterative procedure for both training and validation with the LSTM model.

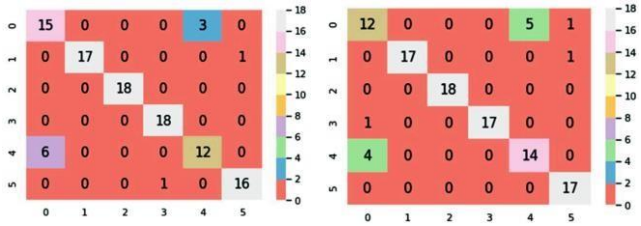


Fig. 11. Confusion matrix with CNN model.

Fig. 12. Confusion matrix with LSTM model.

B. Comparison of Results from CNN and LSTM Models :

Then constructed a classification report for the two models with the classification results achieved. There it was finally concluded that the CNN model shows far better accuracy terms compared with the LSTM model (Table III).

TABLE III. RESULTS COMPARISON BETWEEN CNN AND LSTM MODELS

Model	CNN Model	LSTM Model
Accuracy	99.53%	84.71%
Average Precision	94%	77%
Average Recall	93%	79%
Average F1-score	93%	76%

Divide & Conquer-based 1D CNN HAR with Test Data Sharpening:

As outlined in Figure 2, our approach conducts two-stage activity recognition by introducing test data sharpening in the middle of the two stages at prediction time. In this section, first describe the identification of abstract activities required for first-stage HAR, and then explain methods for test data sharpening and selection of relevant sharpening parameter values. 3.1. **Identifying Abstract Activity & Building 1st-Stage Classifier**

At the outset, tried to build a single sophisticated activity recognition classifier for multi-class HAR, but during the research process and discovered that for certain pairwise activity classes, there were no misclassified instances. This finding could be easily visualized using an activity recognition confusion matrix. Figure 3 shows an example confusion matrix of decision tree classifier on 6-class HAR; the six activity classes are those given in Figure 1. The rows indicate the actual activity classes and the columns indicate the predicted classes. As shown, for some pairwise activity classes, there are no misclassified instances; i.e., the bottom left and the top right 3 × 3 submatrices all contain zeros. Meanwhile, the two red squares drawn over the confusion matrix contain both the correct and misclassified instances, and these demarcated activity classes can be transformed into abstract activity classes. The abstract activity classes are then utilized as target labels for building a binary classifier for first-stage HAR. In the case of Figure 3 confusion matrix and converted walk, WD, and WU classes (Figure 3 top left) to dynamic class, and sit, stand, and lay classes (Figure 3 bottom right) to static class. It is important to note that recognition performance of first-stage binary classifier will determine the upper limit of the overall activity

recognition accuracy since the second-stage activity classifiers, however perfect, will not affect the initial binary classification accuracy. While the divide and conquer approach is advantageous in that the complex (or many-class) classification problem is reduced to multiple simple (or less-than-many-class) classification problems, the approach requires that all classifiers reach reasonably good accuracy; a concerted effort among all simple classifiers are needed. Hence, whether to exploit the divide and conquer approach should be decided when at least the first-stage binary classification accuracy is reasonably high. In our experiments, the activity recognition accuracy of two first-stage binary classifiers on two benchmark datasets were 100%. Figure 3. Confusion matrix of decision tree classifier on 6-class HAR. For some pairwise activity classes, there are no misclassified instances as indicated by the positions with zeros in the confusion matrix. 3.2.

Building 2nd-Stage Classifier :

Once build a high-accuracy first-stage binary classifier and proceed with the learning of the second-stage individual activity recognition classifiers. The detailed design and implementation of our second-stage 1D CNN models are described in Section 4. And finish building the best possible classifiers for the second-stage individual activity recognition and move to test data sharpening.

Sharpening Test Data:

Figure 4 shows the overall signal sharpening process applied to single test data. The train and test data for HAR are constructed in various formats, but usually they are formatted either as activity signal time series data in the form of matrices or as activity signal feature vectors. In the case of Figure 4, Assume that the test data is defined as an m-dimensional feature vector carrying various activity signal features. The test data sharpening is proceeded as follows: Firstly, a Gaussian filter is applied to the test data to remove minor features (see Figure 4 1 and Equation (1) below). The Gaussian filter has the effect of attenuating high frequency signals, and the degree of attenuation is determined by the σ parameter (Figure 4 2). As a result, a de-noised test data can be obtained. Next, the de-noised vector is subtracted from the original test data vector to produce a fine detail vector (Figure 4 3 and Equation (2)). The fine detail vector is then scaled by some scaling factor α (Figure 4 4) before being added to the original test data vector (Figure 4 5 and Equation (3)) to produce a sharpened test data.

5. CONCLUSION

In this paper, presented a CNN model and a LSTM model with 99.593% accuracy and 84.71% accuracy respectively for 6 daily life activities with the WISDM dataset. Use of Conv2D layers for CNN, Dropout regularization and using perfect model hyper parameters in the networks of the two models has made them fast and robust in terms of speed and accuracy. As further works, authors present an idea of using this presented Human Activity Recognition framework as a solution for a smart childcare or eldercare monitoring system based on IoT technologies. Also, it will be a perfect task if it can generate our own dataset with the use of appropriate sensors and applications for a defined number of frequent activities people are performing in day to day lives. This research area seems having multiple advanced applications with Deep Learning applications in near future. In addition, as future works authors suggest the application of reinforcement learning paradigm on the domain of activity recognition and classification.

REFERENCES

[1] L. Alpoim, A. F. da Silva, and C. P. Santos, “Human Activity Recognition Systems: State of Art,” in 2019 IEEE 6th Portuguese Meeting on Bioengineering (ENBENG), Lisbon, Portugal, Feb. 2019, pp. 1–4, doi: 10.1109/ENBENG.2019.8692468.

[2] S. Oniga and J. Suto, “Human activity recognition using neural networks,” in Proceedings of the 2014 15th International Carpathian Control Conference (ICCC), Velke Karlovice, Czech Republic, May 2014, pp. 403–406, doi: 10.1109/CarpathianCC.2014.6843636.

[3] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, “Activity recognition using cell phone accelerometers,” SIGKDD Explor. Newsl., vol. 12, no. 2, pp. 74–82, Mar. 2011, doi: 10.1145/1964897.1964918.

[4] Murad and J.-Y. Pyun, “Deep Recurrent Neural Networks for Human Activity Recognition,” Sensors, vol. 17, no. 11, p. 2556, Nov. 2017, doi: 10.3390/s17112556.

[5] Jobanputra, J. Bavishi, and N. Doshi, “Human Activity Recognition: A Survey,” Procedia Computer Science, vol. 155, pp. 698–703, 2019, doi: 10.1016/j.procs.2019.08.100.

[6] P. Kuppusamy and C. Harika, “Human Action Recognition using CNN and LSTM-RNN with Attention Model” International Journal od Innovative Technology and Exploring Engineering(IJITEE), vol.8, Issue 8, pp.1639-1643, 2019

[7] Y. Chen, K. Zhong, J. Zhang, Q. Sun, and X. Zhao, “LSTM Networks for Mobile Human Activity Recognition,” presented at the 2016

[8] International Conference on Artificial Intelligence: Technologies and Applications, Bangkok, Thailand, 2016, doi: 10.2991/icaita16.2016.13.

[9] C. Hofmann, C. Patschkowski, B. Haefner, and G. Lanza, “Machine Learning Based Activity Recognition To Identify Wasteful Activities

[10] In Production,” Procedia Manufacturing, vol. 45, pp. 171–176, 2020, doi: 10.1016/j.promfg.2020.04.090.

[11] L. B. Marinho, A. H. de Souza Junior, and P. P. Rebouças Filho, "A New Approach to Human Activity Recognition Using Machine Learning Techniques," in Intelligent Systems Design and Applications, vol. 557, A. M. Madureira, A. Abraham, D. Gamboa, and P. Novais, Eds. Cham: Springer International Publishing, 2017, pp. 529–538.

[12] T. Zebin, M. Sperrin, N. Peek, and A. J. Casson, "Human activity recognition from inertial sensor time-series using batch normalized deep LSTM recurrent networks," in 2018 40th Annual International

[13] Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Honolulu, HI, Jul. 2018, pp. 1–4, doi: 10.1109/EMBC.2018.8513115.

[14] Wikipedia, "List of Python software,"2020. [Online]. Available: https://en.wikipedia.org/wiki/List_of_Python_software. [Accessed: 20-Sep- 2020].