Computers and Electrical Engineering 101 (2022) 107946

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

Parkinson and essential tremor classification to identify the patient's risk based on tremor severity ☆

Jigna J. Hathaliya [a], Hetav Modi [a], Rajesh Gupta [a], Sudeep Tanwar [a,*], Priyanka Sharma [b], Ravi Sharma [c]

[a] Department of Computer Science and Engineering, Institute of Technology, Nirma University, Ahmedabad, India
[b] Vice President Projects - Artificial Intelligence, Samyak Infotech and AI Advisor, Orbit Pharma, Gujarat, India
[c] Centre for Inter-Disciplinary Research and Innovation, University of Petroleum and Energy Studies, Dehradun, 248001, India

ARTICLE INFO

Keywords:
Parkinson's disease
Essential tremor
Parkinson tremor
Tremor severity
Convolutional neural network

ABSTRACT

Parkinson's disease (PSD) and essential tremor (ET) are oscillatory and rhythmic movements in the human body with similar characteristics and becomes challenging to identify it accurately. Thus, the chances of misdiagnosis are high. Researchers employed machine learning (ML) algorithms to accurately classify ET and PSD patients. This requires manual feature extraction that, without knowing their importance in prediction purposes, can be mitigated with automated feature engineering using deep learning (DL). So, in this paper, we propose a convolutional neural network (CNN)-based classification model with seven hidden layers and different filter sizes for the accurate classification of PSD and healthy control (HC) subjects. A flatten layer converts three-dimensional data to one-dimensional Tensor flow. Finally, the dense layer outputs the classification of PSD and HC patients based on tremor intensity to identify the PSD patient's risk at an early stage. It outperforms the traditional models with 92.4% accuracy of tremor classification.

1. Introduction

PSD tremor and ET are movement disorders associated with the appearance of tremor [1]. Tremor is a rhythmic, oscillatory, and involuntary movement in human body parts. Tremor distributes in ET and PSD tremor, where ET appears when the patient is performing any muscle activity. PSD tremor occurs when the patient does not perform any activity and body parts are in a resting position [2]. ET can see in older adults an age 60 and above. ET is a progressive movement disorder caused by the motor and non-motor symptoms [3]. Both disorders share similar medical symptoms like bradykinesia, tremor, rigidity, dystonia, and gait impairments, whereas non-motor symptoms like sleep disorders, depression, anxiety, and cognitive disfigurement [4]. Due to both disorders' similar clinical characteristics, it is sometimes challenging to identify the tremor and improve its diagnosis's quality [5].

Nowadays, Neurologist measures the tremor severity of ET and PSD tremor patients during hospital visits. Neurologist performs UPDRS III assessment, in which patients ask to perform the given tasks such as armrest, arm extension, and touch the nose with their index finger [6]. Moreover, a single examination frequently misses the entire spectrum of tremors that PSD subjects experience in their routine [7]. It affects the patient's quality of life and doctors can improve it by delivering optimal care to their patients using remote assistance/monitoring. The real-time data is collected from wearable sensors and transferred to the mobile via Bluetooth

☆ This paper is for special section VSI-aihc. Reviews were processed by Guest Editor Dr. Deepak Gupta and recommended for publication.

* Corresponding author.
E-mail addresses: 19ftphde36@nirmauni.ac.in (J.J. Hathaliya), 19bce078@nirmauni.ac.in (H. Modi), 18ftvphde31@nirmauni.ac.in (R. Gupta), sudeep.tanwar@nirmauni.ac.in (S. Tanwar), drpriyankasharma.ai@gmail.com (P. Sharma), ravisharmacidri@gmail.com (R. Sharma).

https://doi.org/10.1016/j.compeleceng.2022.107946
Received 4 November 2021; Received in revised form 15 March 2022; Accepted 22 March 2022
Available online 22 April 2022
0045-7906/

interface. Mobile devices obtain patient's data using the android API. Moreover, this information is being shared with the cloud server via an open communication channel. Further, this information is being collected and forms a dataset that aids researchers working in the same domain. It improves the diagnosis quality of ET and PSD tremor patients.

Some research studies [1,8] describes tremor monitoring with medical assessment using EMG and accelerometers in which the doctor measures tremor frequency, intensity, and activation pattern to identify the tremor severity of patients [1]. Medical assessment of activation patterns is recently performed on EMG measurements. EMG can extract the electrophysiological parameters of PSD tremor and ET and generate two types of activity patterns, such as asynchronous and synchronous burst transfer. In ET patients, synchronous patterns are observed, and asynchronous patterns are being identified in PSD tremor patients during burst transfer. These synchronous or asynchronous patterns measured using an EMG produce results on a qualitative basis that may lead to misdiagnosis due to the phase-shifting between signals based on the operator's interpretation.

For accurate classification, authors in [8] designed a classification scheme in which they classify PSD and ET tremors using ML algorithms [9,10]. It extracts features manually and the chances are that it can miss some important features. This can be eliminated with the adoption of DL algorithms where the feature extraction process is automatic that enhance the overall classification accuracy [11]. Several recent studies in recent years have incorporated various DL techniques for the diagnosis of PSD [8,12]. Authors in [12] used multi-scale dynamics to estimate the hand tremor and also for the classification of ET and PSD tremor. Their scheme achieved an 80% accuracy of classifying ET from PSD tremor. Later, [8] presented a convolution long short term memory (LSTM) model for tremor classification. The authors have integrated LSTM with CNN and achieved an accuracy of 90%. The problem with the aforementioned works is that they have not identified the patient's risk based on tremor severity. They have also not measured the progression of disease based on tremor severity. The progression of disease helps to identify the risk level on an early basis and aid patients in taking some preventive measures early.

Motivated by this, in this paper, we propose a CNN-based model to classify PSD and HC subjects. Initially, PD-BioStampRC21 data is being collected from IEEE data port [13]. It contains a PSD accelerometry dataset of 5 wearable sensors for both PSD and HC subjects. The data is being collected using lightweight MC 10 biostamp RC sensors. The basic requirement of the model is to convert the raw data and normalize it before feeding to the CNN model. We applied data pre-processing to normalize and scale in a similar range using a standard scalar function for data normalization. Then, this data feed into the CNN model for accurate classification of PSD tremor and ET. The proposed CNN model consists of 7 layers with different filter sizes. CNN takes the convolutional 2D layer as an input layer and generates the output at the dense layer. In model training, all layers are connected in which output of the first layer feed as an input layer to the next layer. A Flatten layer converts 3D data into a 1D Tensor flow. The output of the flatten layer is used as an input to the next connected layer. In the end, the dense layer produces a promising result that classifies ET and PSD tremors accurately from the entire dataset.

Moreover, Adam optimizer is used for model optimization during the compilation process. Finally, we evaluated the proposed model with a 0.0025 learning rate and calculated a loss using categorical cross-entropy, accurately predicting each sample present in the dataset. The proposed model gives a promising classification of PSD and HC subjects. The proposed model outperforms with 92.4% accuracy, which is better as compared to [8,12] schemes.

1.1. Motivation

It is challenging to differentiate ET and PSD tremors due to their similar characteristics of movement disorders. ET is generally seen at the age of 60 and above, while the PSD tremor is seen during the rest position of PSD patients. Authors in [14] used ML techniques to detect ET and PSD tremors individually using wearable and accelerometer sensors. ML causes a manual handcrafted feature extraction issue, which leads to a lower model's performance. Later, many authors addressed this issue in their research articles [15–18]. They proposed DL-based techniques that give automatic feature extraction instead of manual feature extraction. To solve the feature extraction issue of ML models, we have used the CNN algorithm to classify the PSD and HC subjects based on PSD tremor and ET. This classification helps to know the patient's tremor severity and recognize the patient's risk level.

1.2. Contributions

Following are the main contributions of the paper:

- We propose a CNN-based scheme to differentiate the ET and PSD tremor patients using three-axis accelerometer data. The DL approach improves the model's performance, improving patient diagnosis accuracy.
- The tremor classification helps to know the tremor severity of the PSD patients and identify the risk level based on a probability function. This aid to see the level of disease progression of PSD patients and taking some preventive measures to lower the tremor severity of the PSD patient.
- The performance of the proposed model is evaluated using various matrices such as accuracy and loss and compared with other schemes.

1.3. Organization

The rest of the paper organized as follows. Section 2 presents the related work. Section 3 describes the system model and problem formulation. Section 4 elaborate the proposed architecture and algorithm. The performance evaluation is presented in Section 5. In the end, Section 6 concludes the paper.

Table 1
Comparative analysis of various state-of-the-art works for tremor detection using wearable sensors.

Author	Year	Objective	Techniques/ Models	Accuracy	Pros	Cons
Hssayeni et al. [7]	2019	Designed an LSTM and GB classification scheme to differentiate ET and PSD tremor using wearable sensors	LSTM and GB for held out and leave one out testing	Correlation r = 0.93	Predict the total rest and action tremor accurately	Complex model
Moon et al. [14]	2020	Proposed a classification scheme to detect PSD and HC subject using ML algorithms	NNA, kNN, GB, SVM, DT, and LR	F1-score: logistic regression: 0.53, neural network: 0.61, GB:0.59, RF: 0.56, SVM: 0.55, decision tree: 0.53, KNN: 0.49	Identify tremor by checking gait and balance characteristics	Low performance
Oktay et al. [8]	2020	Distinguish ET and PSD tremor from 3 axis hand tremor data using LSTM network	Convolutional LSTM network	90% Accuracy	Correctly classified ET and PSD tremor using LMC controller	Complex model
Su et al. [12]	2020	Identified various effects on ET and PSD tremor on hand tremor model for accurate diagnosis of PSD and HC	Multiscale dynamics	80% Accuracy	Accurate prediction of disease severity on hand tremor	Misdiagnosis of disease.
The proposed approach	–	Proposed a CNN-based classification scheme to differentiate the ET and PSD tremor and classify PSD and HC subjects using wearable sensors	CNN-based scheme	92.4% Accuracy	Accurate diagnosis of patient and measures the tremor severity to identify the risk level of PSD patient based on probability function	–

2. Related work

This section presents several studies of tremor analysis and tremor detection of PSD and ET using wearable sensors such as accelerometers, smartwatches, and interial sensors [15–18]. Authors in [14] used ML algorithm to analyze and detect PSD and ET tremors based on gait and balance characteristics. They have used wearable sensors to collect the three-axis tremor data. They used various ML algorithms such as random forest, neural network, decision tree, gradient boosting (GB), K nearest neighbor, and logistic regression and evaluated the performance using precision, recall, accuracy, and F1 score. Their model designed an efficient tool for diagnosis that assists doctors in the decision-making process. Further, it also helps to know the patient's disease progression. Later, [7] presented a tremor classification scheme based on wearable sensors in which sensors were used to collect measurements from the ankle and wrist. The model train based on that samples and gives the superior classification result. They used LSTM and GB algorithms to measure the tremor severity of any patient. They have used 24 PSD patients and tested them using a gyroscope sensor. They evaluated their system with two approaches: held out and left one out testing. The scheme achieves 84%, 96% accuracy in held-out testing and leave one out testing performs 77% and 93% accuracy of total rest and action tremor.

Then, the authors in [8] differentiate both ET and PSD tremors using LSTM networks. The Leap Motion Controller (LMC) collects 3D data for classification. The model uses 40 subjects and is tested with 90% accuracy in combined tremor analysis. In this scheme, the medical assessment of ET and PSD tremor using LMC without identifying features such as frequency and power spectral density. They correctly identify the tremor severity of the patient. Later, Su et al. [12] designed a hand tremor effects detection scheme. This effect uses multiscale dynamics to monitor the hand tremor effects in movement disorders like ET and PSD tremor. In this scheme, accelerometer sensors data helps quantify the complexity of both tremors. This scheme monitors the tremor duration and severity of the disease, which helps to differentiate ET and PSD tremors. The model was trained with 48 subjects and evaluated with 77% accuracy.

We have used the CNN algorithm to classify PSD and HC for an accurate diagnosis of ET and PSD tremor. Initially, accelerometer sensors gather patient data to measure the right and left-hand tremor data in 3 axes. Then the raw data is normalized using a standard scalar function in which standardization of data using mean and standard deviation scale the data in 0 to 1 range. Then we trained the model with training samples that consist of various layers such as convolution, drop out, flatten, and dense layers trained with different kernel sizes. The dense layer generates the classification result of ET and PSD tremor. Table 1 shows the comparative analysis of existing schemes over parameters such as the objective, algorithm, pros and cons.

3. System model and problem formulation

This section presents the system model and problem formulation with various mathematical equations.

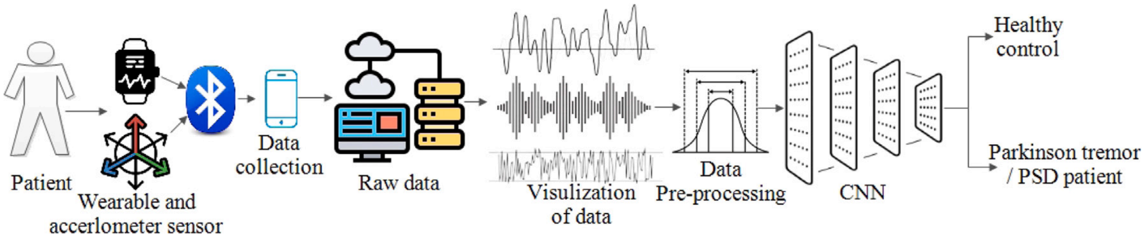


Fig. 1. System model.

3.1. System model

Fig. 1 shows the system model with three distinguished phases such as data collection, preprocessing, and model training and validation phases. A wearable or accelerometer sensor has been used to collect the tremor data from the left and right hands of the patient. The data was collected over X, Y, and Z directions. Further, this data is transferred to a smartphone using a Bluetooth communication channel. The data is stored in a CSV file over the cloud server to access the data as per the need. This raw data visualization is present in 3D, in which patient activity, tremor intensity, and tremor compensation are seen in terms of frequency. After data collection, data preprocessing is an essential requirement to derive meaningful data from the entire dataset. The proposed model uses a standard scalar function to standardize input values by removing the mean and scaling all the features into the unit variance to normalize the data. These feature values are updated by calculating the scaling score. We first calculate the mean using the following equation to calculate the standardization.

$$\mu = \frac{1}{N} \sum_{k=1}^{N} X_k \tag{1}$$

Eq. (1) presents the mean μ of the input data and X_k is the k^{th} sample of the data and N describes the number of samples. After that, we compute the standard deviation as follows:

$$\sigma = \sqrt{\frac{1}{N} \sum_{k=1}^{N} (A_i - \mu)^2} \tag{2}$$

where σ denotes the standard deviation. N denotes the size of the sample, A_i denotes each value of the sample, and μ denotes the mean. After that, we standardize the values using the following equation:

$$S = \frac{A - \mu}{\sigma} \tag{3}$$

Eq. (3) is the result of standardization using S, where A is the value of each sample, μ is mean, and σ is a standard deviation.

After preprocessing, the system model is trained using training examples with seven different hidden layers, where the individual layer is composed of varying kernel sizes. First, we have performed forward propagation in which we take the convolution 2D layer as an input layer and the output of that layer feed into the next layer. Thus, this procedure follows with dropout and convolutional 2D layers. After that, flatten layer transfers 3D data into 1D tensor flow in which we flatten the result from the preceding layer and produces the final output in a dense layer. The mathematical formulation of the model is present as follows:

$$L_1 = D \times F_s \tag{4}$$

Eq. (4) shows the computation of layer 1 with L_1 notation, where D is the data and F_s is the filter size. Each individual layer consists of different filter and kernel sizes. The output of each layer feeds as an input to the next layer with such dimensions, which is calculated as follows:

$$Dimension(n_i - F_s + 1) \tag{5}$$

Eq. (5) presents the output dimension, and data dimension denotes with (n, n) and filter denotes with (F_s, F_s). Further, a system model is trained with Rectified Linear Unit (ReLU) activation function to reduce the vanishing gradient problem. The ReLU activation function is applied to the output vector L_1 and generates the output, which can be computed as:

$$ReLU(s) = max(0, s) \tag{6}$$

Eq. (6) shows the ReLU activation function, where s denotes the sample. Then, we apply this function for a next layer of CNN which can be defined as:

$$L_2 = ReLU(L_1) \tag{7}$$

Eq. (7) describes the output of each layer which is fed into the next layer, and L_1 is the preceding layer dimension of filter size and data. Each layer has been assigned with some weights and bias which is computed using the following equation:

$$L_3 = (W_e * L_2) + b_i \tag{8}$$

Eq. (8) calculate the result L_3 containing weight W_i and L_2 is the preceding layer output dimension and bias value denotes with b_i. At the flatten layer, data is converted into 1D tensor flow data then dense layer generates the final output with softmax activation function σ, which is mathematically computed as following:

$$\sigma \vec{Z}_i = \frac{e_j^z}{\sum_{k=1}^{N} e_k^z} \tag{9}$$

In Eq. (9) presents softmax activation function in which input vector denoted with $\vec{Z}_i$, standard exponential function for input vector denoted with e_j^z, number of classes in the multi-class classifier is denotes with N and standard exponential function for output vector is denote with e_k^z.

To enhance the performance of the model, we back propagate the model in which the model updates the parameters in a way that improves the predictions and performance of the model. The new parameter value is update as below:

$$new_p = old_p - (Le_r * Gr_p) \tag{10}$$

Eq. (10) describes the value of new parameter n_p old_p contains the value of old parameter. The learning rate and gradient value are represented with Le_r and G_p.

After training, the model loss is calculated using the categorical cross-entropy which is present as below:

$$CCE = -\sum_{j}^{C_l} m_i \log(n_i) \tag{11}$$

Eq. (11) presents the entropy which denotes with CCE, where m_i and n_i are ground-truth and CNN score for individual class j in classes C_l.

PSD tremor and ET patients have shared similar medical symptoms, so it becomes difficult to identify the category of tremor manually. It leads to a misdiagnosis of the patient's report. For an accurate and improved diagnosis of the tremor category, we have used the CNN model to maximize the accuracy and reduce the loss of the model, which helps to improve the accuracy of the diagnosis report. This differentiation also helps to know the risk level of the PSD patient based on tremor severity and the disease progression of the PSD patient. As per the discussion, the objective function for the proposed scheme is specified as follows:

$$\rtimes \xrightarrow{M} ax(CNN) \forall DR_{\mathrm{PSD}} \rightarrow Accuracy \tag{12}$$

In the Eq. (12), Diagnosis report of PSD patient denotes with D, R_{PSD}, and The output of the objective function denoted with the o.

4. The proposed approach

This section describes the proposed approach for tremor classification. The tremor data was initially collected in 3-axis using an accelerometer or wearable sensor. Further, this data is shared with a smartphone using Bluetooth as a communication channel. After that, the data is stored in the cloud server that can access as needed. After that, the tremor data is converted into a visualized form to see data over the X, Y, and Z directions. The data should be required to be pre-processed before feeding them into the model for training. Fig. 2 shows the hierarchical clustering dendrogram and a subset of PSD tremor and HC data. The dendrogram has been generated using the equal number of readings collected for PSD tremor and ET from the dataset. The clustering dendrogram is an unsupervised learning technique that clusters the dataset into different classes based on similarity. The distinction of two classes using the unsupervised technique affirms the accelerometer readings obtained from the sensors. It is a crucial method to aid the identification of the PSD tremor from the dataset.

4.1. Dataset description

PD-BioStamp dataset [13] contains PSD accelerometry data of five wearable sensors of both PSD and HC participants. Data is gathered using lightweight MC 10 BioStamp RC sensors. Each participant collected data using the five sensors attached to their trunk, left anterior thigh, right anterior thigh, left anterior forearm, and right anterior forearm. We use only the left anterior forearm readings to train our model. The dataset contains each measurement of triaxial accelerometry readings relative to the sensor coordinate system. It includes the values of acceleration in the X, Y, and Z directions, considering the importance of gravitational constant g = 9.8 m/s^2. The readings are taken consistently at regular intervals and reported in milliseconds. The dataset also contains information about the status of the participants' medication.

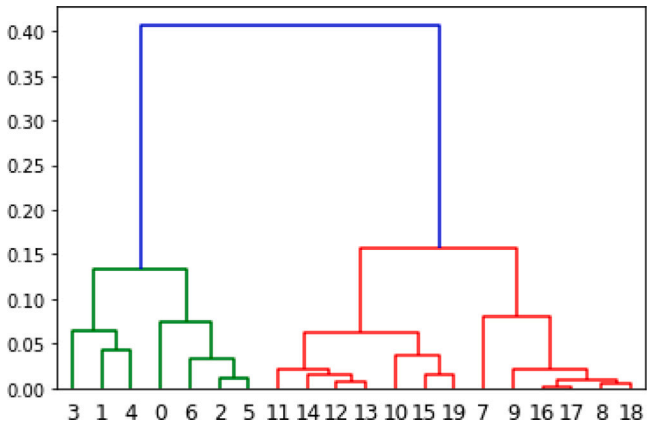


Fig. 2. Hierarchical clustering of PSD and HC data.

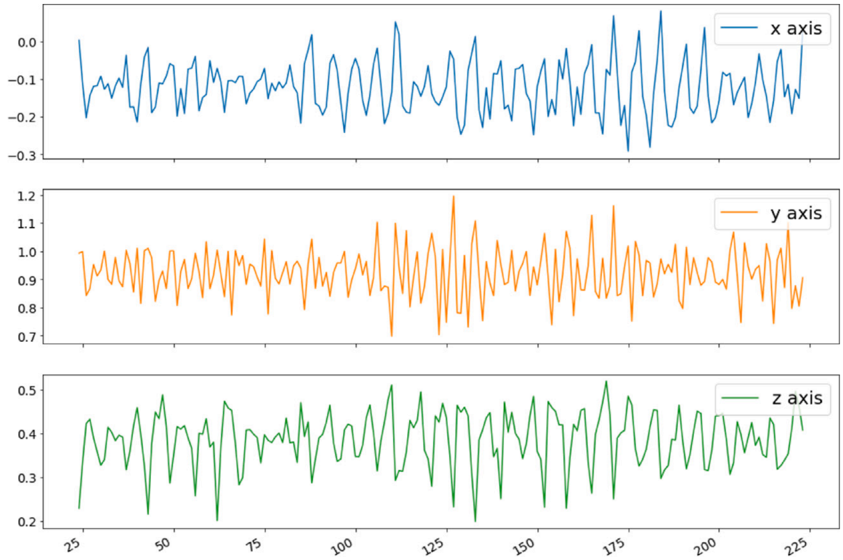


Fig. 3. PSD Patient activity in X, Y, Z directions collected using sensors.

4.2. Data pre-processing

We gathered continuous time-series data of triaxial accelerometry for the left anterior forearm for both PSD and HC participants. We concatenated 30,000 readings for each patient along with their status, PSD or HC. The concatenated data is checked for null values. The rows with null values are dropped. The status of a patient is encoded using a label encoder. PSD is encoded as one, and HC is represented as 0. Label encoding is followed by scaling the data. We standardize the data present into a fixed range. Scaling is used to reduce the influence of higher magnitude readings on the classification output. A standard scaler is used to standardize input values by removing the mean and scaling all the features to unit variance. Feature values are updated by calculating the scaling score. After scaling, the frames are prepared, consisting of 200 rows of overlapping data. So, in this case, the network can learn from the small dataset. Overlapping data ensures that the model can learn from the continuous data and make inferences about whether the data belongs to the patient with PSD tremor or ET. We assign the mode of the label to each frame so that it gives a reasonable estimate of which class that particular frame belongs to. Taking the mode of the label allows the considerations of cases where the frames may have two different classes due to the concatenation of data.

Fig. 3 shows the PSD participant activity as detected using sensors. We take readings for a time frame of 200 timestamps that give insight into the data's time-series nature while taking only a small portion of the data. Fig. 4 shows the HC participant activity as detected using sensors. Fig. 5(a) shows the autocorrelation or the serial correlation for the readings. It demonstrates the similarity among the observations as a function of delay. They help to find the repeating patterns in time series data. We have plotted the readings with a lag of 100 timestamps across the x-axis readings obtained from the accelerometer. The partial correlation function for the same set of readings is plotted in Fig. 5(b). The partial autocorrelation function represents the partial correlation of the

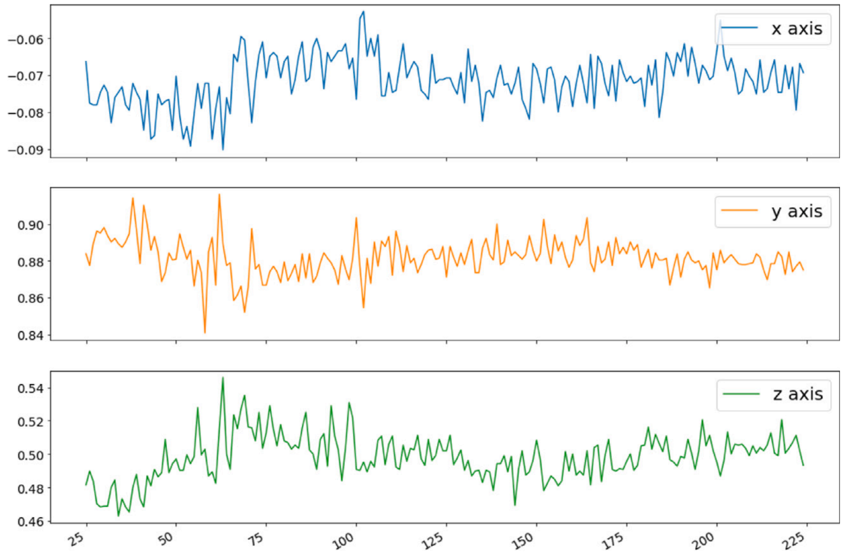
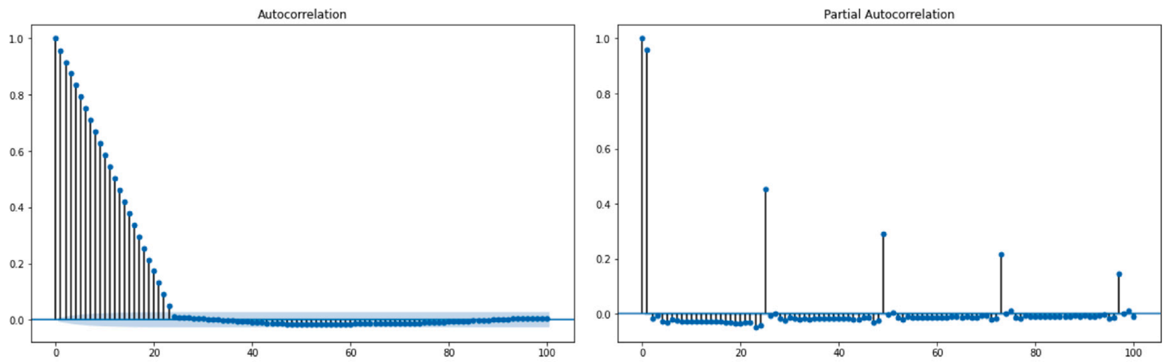


Fig. 4. HC Patient activity in X, Y, Z directions using sensors.



(a) (b)

Fig. 5. (a) Autocorrelation plot. and (b) Partial autocorrelation plot.

readings with its own lagged reading values. It helps to identify the extent of lag in the time series readings. By evaluating the Figs. 5(a), 5(b) we have been able to identify a suitable value of the window size of recordings taken as hyperparameter for the proposed model as discussed in Algorithm 1.

4.3. Proposed model

Fig. 6 shows the proposed model in which the model is trained using CNN with seven layers with different filter sizes and dimensions. The convolutional 2D layer took input and processed tri-axial readings obtained from the sensors. The convolutional layer consists of a set of filters or kernels, whose parameters will be learned while training the model [19]. The size of kernels are usually smaller than the actual input size, and each filter convolves over the filter map and generates the activation map. The convolutional layer helps to decrease the number of parameters to be learned and improves the overall efficiency compared to a fully connected deep network. The convolutional 2D layer has 16 units and a stride of 2 × 2. The stride refers to the number of shifts while convolving the kernel over the input matrix. In the end, the dropout layer is being used to reduce the over-fitting of the model. This layer uses a random distribution probability to set the hidden units' outgoing edges as 0 based on the specified probability. The dropout layer provides a 0.1 probability to ensure the validity of the training model. Another convolutional 2D and dropout layer set contains 32 units in the convolutional layer, 2 × 2 stride, and a 0.2 dropout probability. Afterwards, the flatten layer converts all 3D data inputs into a 1D layer, followed by a fully-connected layer with 64 units. The fully connected layer is followed by another dropout layer with a 0.5 dropout probability. This probability helps to connect the final output dense layer to produce a binary classification.

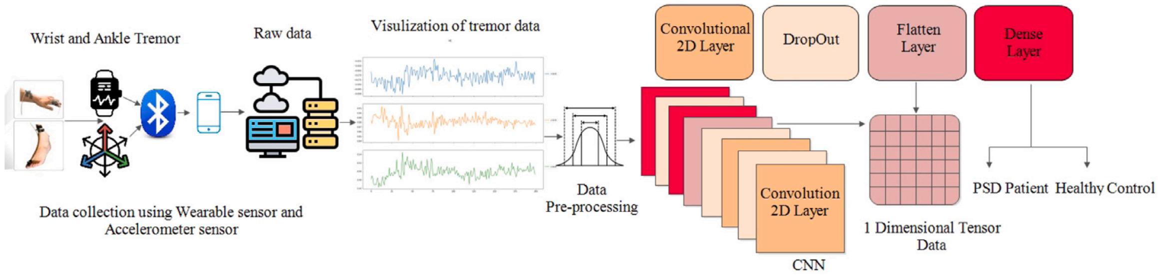


Fig. 6. The proposed model.

Each convolutional 2D and dense layer uses the ReLU activation function to introduce nonlinearity, and the final output layer uses a softmax function as activation to represent the output in terms of the specified probability of the class. This probability also helps to know the tremor severity of the patient and identify the patient's risk. Based on this risk, the patient can take preventive measures to improve their health. After model training, we evaluate the trained model using a test dataset and evaluate it on various metrics such as accuracy and loss.

4.4. Algorithm

Algorithm 1 presents the execution process of the algorithm. The patient's tremor data was initially collected in 3-axis using an accelerometer or wearable sensor. We applied data preprocessing to derive meaningful data from the entire dataset. The standard scalar increases the window size and scales the data from 0 to 1. Moreover, the dataset is split into training and testing input data. The test split is 0.2 of the total data. These preprocessed data are then fed into the CNN model with seven hidden layers using different filter sizes. CNN produces the result at a fully connected dense layer and applies model optimization using adam optimizer. Adam is a replacement optimization algorithm of stochastic gradient descent. This algorithm is used to train a DL model. Adam optimizer involves a combination of two gradient descent methodologies. The first momentum is used to accelerate the gradient descent algorithm by considering the exponentially weighted average of the gradients.

The second is a root mean square prop (RMSProp) that takes the exponential moving average. The trained model was compiled using a 0.0025 learning rate. The learning rate was determined using different learning rate values keeping other hyperparameters constant and analyzing the optimal value of the learning rate for the model performance. The loss of the model is being calculated using categorical cross-entropy. The categorical cross-entropy calculate the loss and uses softmax as the activation function for the final classification output. The proposed model was evaluated using a testing dataset in the model fit procedure. After model training for 30 epochs, the model generates a significant output. The model is evaluated using performance metrics such as accuracy and loss. This algorithm helps to identify the PSD tremor from ET and know the risk level of PSD tremor patients. The patient's risk can be monitored by evaluating and comparing the similarity of the PSD tremor and ET to know the level of disease progression. The time complexity of the preprocessing part of the proposed algorithm is O(n). While determining the complexity of a DL model remains relatively complex, and it varies based on the heuristics involved, the complexity can be defined in terms of the parameters mentioned above like the learning rate, number of weights and connections between the layers. The space complexity deals with the stored weights, biases and intermediate computations during model training.

5. Performance evaluation

This paper proposed a CNN-based approach to differentiate between PSD tremor and ET. ET are common amongst the elderly population and its symptoms mimic the PSD tremor. For correct diagnosis of PSD, the patient gets early medical assistance that is necessary for neurodegenerative diseases. We use the PD-BioStampRC21 dataset and the CNN-based DL algorithm to classify the difference between PSD tremor and ST using a human activity sensor. The dataset contains a total of 32 participants with both PSD and HC. Among these participants, 18 were female, and 14 were male.

Fig. 7 shows the distribution of age for both male and female participants. The average age of the male participant was 64.3 and the average age for the female participant was 66.5. The highest age in the dataset was 84 years and the lowest age was 37. The number of participants in the study with HC was 15 and patients with PSD were 17. Fig. 8 shows the distribution of participants with PSD and HC. A balanced dataset with HC ensures that the output does not lean towards a singular class and ensures high accuracy for the model. We train our CNN model to classify the patients based on the time series data as PSD tremors and HC tremors. After, we compile our model using categorical cross-entropy as the loss function. After tuning the parameter for several different values, we pass the Adam optimizer with a learning rate of 0.0025. After calling the compile function, we fit our model. We train our model for 30 epochs and validate the testing data. 80%–20% split is used to split training and testing data.

Table 2 shows the detailed architecture of the proposed CNN model. Convolutional 2D layers are interspersed with dropout layers. Flatten and dense layers follow to generate the output of the classification. Fig. 9 represents the comparison of the accuracy obtained by different baseline models. We have considered two types of baseline models to compare the proposed scheme. The first

Algorithm 1 Working process of the proposed scheme.

START
Input: Accelerometer sensor data in X,Y and Z direction
Output: Classification of patients as PSD and HC Subject

```
procedure PRE-PROCESSING(:)
    Label encoding: [0,1] ←['PSD', 'HC']
    Scaling : Standard Scaler (X,Y,Z)
    Window Size ←200
    a ←Window size × 4
    b ←Window size × 2
    for i in range (0, length(input), b) do
        X' ←X[i: i + a]
        Y' ←Y[i: i + a]
        Z' ←Z[i: i + a]
        Label ←Mode [i: i + a]
        Train Test Split (test size = 0.2)
        Reshape: (Input required = 3D)
        X_train ←X_train.reshape()
        X_test ←X_test.reshape()
    end for
end procedure
procedure MODEL TRAINING(:)
    for < i in 7 Layers > do
        Apply Filter (2 × 2)
        Apply Activation
        Apply Dropout
    end for
    Dense layer generates the output
end procedure
procedure Compile(:)
    Adam ←Optimizer
    0.0025 ←Learning_rate
    Categorical_Cross_entropy ←Loss
end procedure
procedure ModelFit(:)
    X ←X_train
    Y ←Y_train
    Validation ←X_test, Y_test
    30 ←epochs
end procedure
```

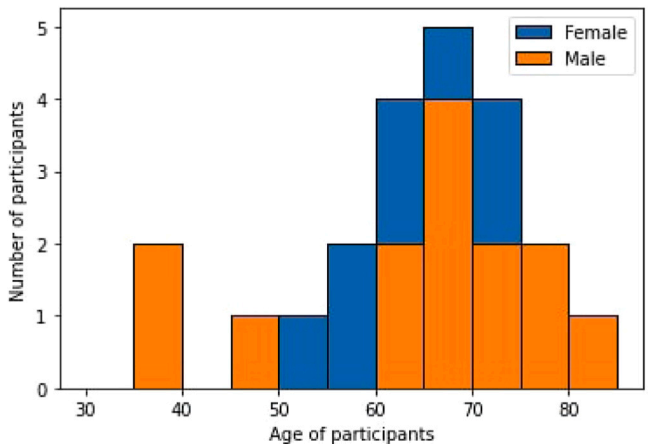


Fig. 7. Age distribution of participants based on their gender.

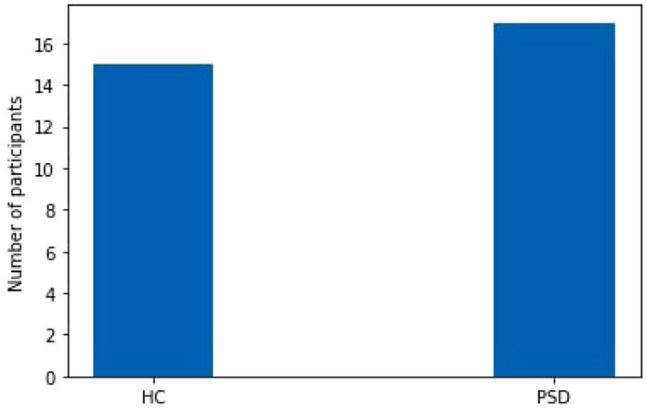


Fig. 8. Distribution of participants with HC and PSD in the dataset.

Table 2
Parameters of the proposed model.

Layer	Layer type	No of units	Activation	No. of strides	Dropout
1	Input	–	–	–	
2	Convolutional 2D layer	16	ReLU	2 × 2	–
3	Dropout	–	–	–	0.1
4	Convolutional 2D layer	32	ReLU	2 × 2	–
5	Dropout	–	–	–	0.2
6	Flatten	–	–	–	–
7	Dense	64	ReLU	–	–
8	Dropout	–	–	–	0.5
9	Dense	2	Softmax	–	–

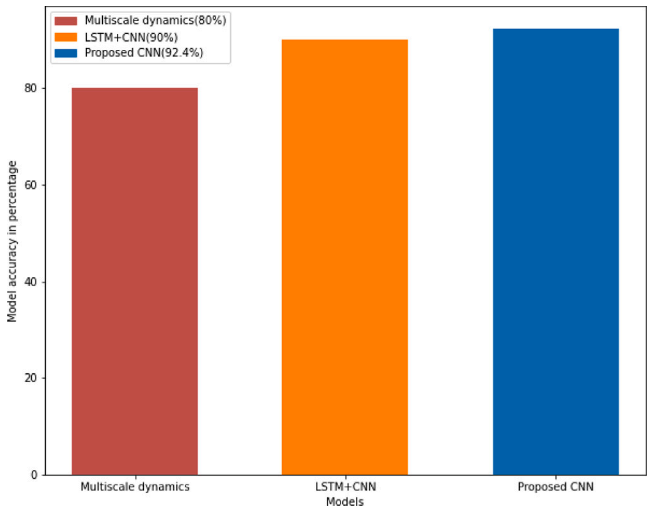


Fig. 9. Comparison of accuracy of different models.

baseline scheme used the multi-scale dynamics as a technique and achieved an 80% accuracy. The second baseline scheme uses a convolutional LSTM technique to accomplish a 90% accuracy. Therefore, we proved that the proposed model outperforms with 92.4% accuracy, which is better than baseline schemes.

Both Figs. 10 and 11 represent the evaluation of the trained model with the same set of parameters obtained after model training with the proposed scheme. The hyperparameters have been fixed after finding a suitable pair of parameters by tuning. All the training inputs are provided during each epoch. The number of epochs is decided by starting from an ad-hoc value and changing those values based on the trade-off between over-fitting and underfitting. The proposed heuristic selects the 30 epochs as an optimal value. If epochs reach lower than 30, the model achieves lower training and validation accuracy. If it comes more significant than 30 epochs, the model can be started to over-fit, where the training accuracy had increased much higher than the validation accuracy. Model

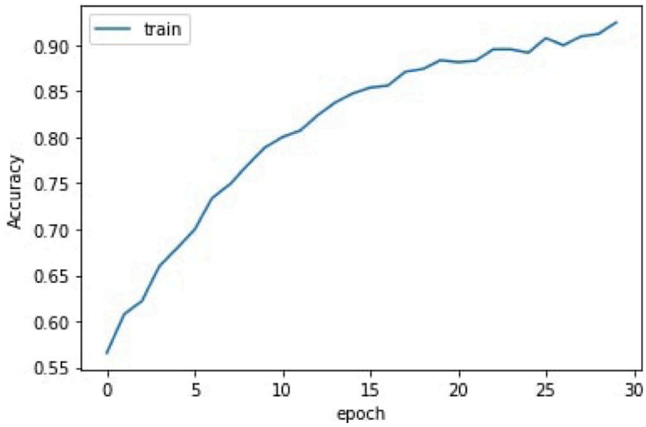


Fig. 10. Accuracy of the proposed model against the number of epochs.

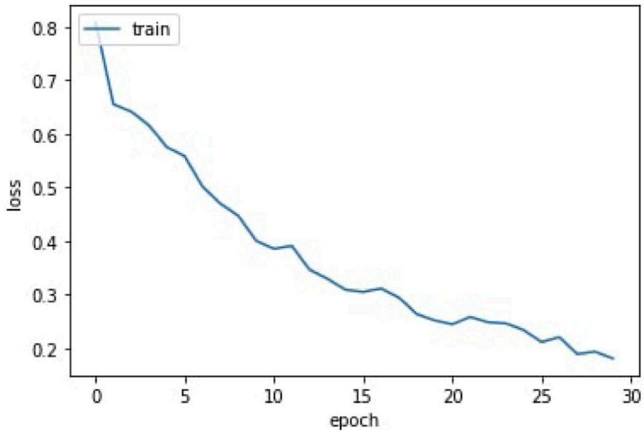


Fig. 11. Loss of the proposed model against the number of epochs.

accuracy is expected to be improved with a larger available dataset. Fig. 10 shows the curve for the accuracy of the CNN model with the numbers of epochs. We see that the accuracy increases as the number of epochs increases. This allows the model to learn and improve its classification accuracy and make better predictions.

The loss of the CNN is calculated using the categorical cross-entropy loss function. We reduced the loss by training the model over 30 epochs to achieve higher accuracy. Fig. 11 presents the loss of the CNN model against the number of epochs. The proposed model achieved an accuracy of 92.4%. We can use this model to aid in diagnosing PSD tremor and ET. PSD has symptoms similar to ET associated with old age. The model predicts the output in terms of probability using the softmax or the normalized exponential function. This helps to identify the patient's risk by assigning different threshold values of probabilities and mapping it against the existing unified Parkinson's disease rating scale (UPDRS) to determine the severity of the disease and assess the patient's risk. The UPDRS scale and patient tremor severity require quicker medical assistance. The proposed model identifies risk based on the UPDRS scale and can notify patients when the risk is beyond a specific value. Studies have found that patients usually get a traditional medical diagnosis of PSD, when more than half of all the dopaminergic neurons may already have been lost [20]. Our proposed approach can identify and notify the patient using the wearable device and it allows early identification of the PSD.

After the classification of ET and PSD tremor, we identify the tremor intensity of the PSD patient and know the tremor severity and risk level of the PSD patient, which helps to know the patient's disease progression. This progression helps patients to take preventive measures to improve their health. For diagnosis, sensor readings from patient activity can assist professional medical diagnosis, especially early in the absence of other PSD symptoms. The proposed model can be extended to readings obtained from other datasets using wearable sensors or devices having the readings available in triaxial directions. The model works on any dataset related to Parkinson or tremor classification and is collected from a validated source. It can also help to test the time series readings available for the participant. The wearable sensor can be attached to the wrist or the foot of the participant to collect the readings.

6. Conclusion

We proposed a DL-based CNN algorithm to differentiate the PSD tremor and ET to identify the risk level of patients based on tremor severity from body movements. CNN facilitates the automatic feature extraction from 3 axis accelerometry data. We have used the PD-BioStampRC21 dataset with 34 subjects containing X, Y, and Z-axis tremor data collected using wearable or accelerometer sensors. First, the data is normalized using standard scalar function and scale data in the range of 0 to 1. The proposed CNN model consists of 7 hidden layers with various filter sizes in which all layers take an input of the preceding layer and generate the output and feed that data to the next layer. Flatten layer generates the 1D tensor flow data from 3D data, and the dense layer generates the classification result. During a model evaluation, the loss of the model is computed using the categorical cross-entropy loss function to enhance the model's performance. The proposed model identifies the tremor severity to know the PSD patient's risk level and disease progression. The model outperforms 92.4% accuracy with an accurate PSD tremor and ET diagnosis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work is supported by Visvesvaraya Ph.D. Scheme for Electronics and IT, India by Department of Electronics and Information Technology (DeiTY), Ministry of Communications and Information Technology, Government of India <MEITY-PHD-2828>.

References

[1] Vescio B, Nisticò R, Augimeri A, Quattrone A, Crasà M, Quattrone A. Development and validation of a new wearable mobile device for the automated detection of resting tremor in Parkinson's disease and essential tremor. Diagnostics 2021;11(2).

[2] Nisticò R, Quattrone A, Crasà M, De Maria M, Vescio B, Quattrone A. Evaluation of rest tremor in different positions in Parkinson's disease and essential tremor plus. Neurol Sci 2022;1–7.

[3] Ma C, Li D, Pan L, Li X, Yin C, Li A, Zhang Z, Zong R. Quantitative assessment of essential tremor based on machine learning methods using wearable device. Biomed Signal Process Control 2022;71:103244.

[4] Zhang B, Huang F, Liu J, Zhang D. A novel posture for better differentiation between Parkinson's tremor and essential tremor. Front Neurosci 2018;12:317.

[5] Sigcha L, Pavón I, Costa N, Costa S, Gago M, Arezes P, López JM, De Arcas G. Automatic resting tremor assessment in Parkinson's disease using smartwatches and multitask convolutional neural networks. Sensors 2021;21(1).

[6] Modi H, Hathaliya J, Obaidiat MS, Gupta R, Tanwar S. Deep learning-based Parkinson disease classification using PET scan imaging data. In: 2021 IEEE 6th international conference on computing, communication and automation (ICCCA). 2021, p. 837–41.

[7] Hssayeni MD, Jimenez-Shahed J, Burack MA, Ghoraani B. Wearable sensors for estimation of Parkinsonian tremor severity during free body movements. Sensors 2019;19(19).

[8] Oktay AB, Kocer A. Differential diagnosis of Parkinson and essential tremor with convolutional LSTM networks. Biomed Signal Process Control 2020;56:101683.

[9] Sheth K, Patel K, Shah H, Tanwar S, Gupta R, Kumar N. A taxonomy of AI techniques for 6G communication networks. Comput Commun 2020;161:279–303.

[10] Thakkar P, Varma K, Ukani V, Mankad S, Tanwar S. Combining user-based and item-based collaborative filtering using machine learning. In: Satapathy SC, Joshi A, editors. Information and communication technology for intelligent systems. Singapore: Springer Singapore; 2019, p. 173–80.

[11] Tong L, He J, Peng L. CNN-based PD hand tremor detection using inertial sensors. IEEE Sens Lett 2021;5(7):1–4. http://dx.doi.org/10.1109/LSENS.2021.3074958.

[12] Su D, Zhang F, Liu Z, Yang S, Wang Y, Ma H, Manor B, Hausdorff JM, Lipsitz LA, Pan H, et al. Different effects of essential tremor and Parkinsonian tremor on multiscale dynamics of hand tremor. Clin Neurophysiol 2021.

[13] Adams JL, Dinesh K, Snyder CW, Xiong M, Tarolli CG, Sharma S, Dorsey ER, Sharma G. PD-BioStampRC21: Parkinson's disease accelerometry dataset from five wearable sensor study. 2020.

[14] Moon S, Song H-J, Sharma VD, Lyons KE, Pahwa R, Akinwuntan AE, Devos H. Classification of Parkinson's disease and essential tremor based on balance and gait characteristics from wearable motion sensors via machine learning techniques: a data-driven approach. J NeuroEng Rehabil 2020;17(1):1–8.

[15] Chen L, Cai G, Weng H, Yu J, Yang Y, Huang X, Chen X, Ye Q. More sensitive identification for bradykinesia compared to tremors in Parkinson's disease based on Parkinson's KinetiGraph (PKG). Front Aging Neurosci 2020;12:356.

[16] Yao L, Brown P, Shoaran M. Improved detection of Parkinsonian resting tremor with feature engineering and Kalman filtering. Clin Neurophysiol 2020;131(1):274–84.

[17] de Araújo ACA, Santos EGdR, de Sá KSG, Furtado VKT, Santos FA, de Lima RC, Krejcová LV, Santos-Lobato BL, Pinto GHL, Cabral AdS, Belgamo A, Callegari B, Kleiner AFR, Costa e Silva AdA, Souza GdS. Hand resting tremor assessment of healthy and patients with Parkinson's disease: An exploratory machine learning study. Front Bioeng Biotechnol 2020;8:778.

[18] Sajal MSR, Ehsan MT, Vaidyanathan R, Wang S, Aziz T, Al Mamun KA. Telemonitoring Parkinson's disease using machine learning by combining tremor and voice analysis. Brain Inform 2020;7(1):1–11.

[19] Mungra D, Agrawal A, Sharma P, Tanwar S. PRATIT: A CNN-based emotion recognition system using histogram equalization and data augmentation. Multimedia Tools Appl 2020.

[20] Murman DL. Early treatment of Parkinson's disease: opportunities for managed care. Am J Manage Care 2012;18(7 Suppl):S183–188.

Jigna J Hathaliya is currently pursuing Ph.D. at Nirma University, Ahmedabad, Gujarat, India. Her research interests are Machine Learning, Deep Learning, Blockchain Technology, and Network Security.

Hetav Modi is currently pursuing a bachelor's degree at Nirma University, Ahmedabad, Gujarat, India. His research interests are Machine Learning, Deep Learning, Blockchain Technology, and Network Security.

Rajesh Gupta is a Full-Time Research Scholar in the Computer science and Engineering Department at Nirma University, India, under the supervision of Dr. Sudeep Tanwar. He has authored/co-authored some publications in SCI Indexed Journals and IEEE ComSoc sponsored International Conferences. His research interests include blockchain and D2D communication.

Sudeep Tanwar (M'15, SM'21) is working as a Professor at Nirma University, India and was visiting professor at Jan Wyzykowski University in Polkowice, Poland and the University of Pitesti in Pitesti, Romania. His research interests include WSN, blockchain, fog computing, D2D communication, and smart grid. He contributed more than 270 research papers in leading journals and conferences and edited/authored 13 books published in leading publication houses. He is also serving the editorial boards of COMCOM-Elsevier, IJCS-Wiley, Cyber Security and Applications- Elsevier, Frontiers of blockchain, and SPY, Wiley. He is also leading the ST Research Laboratory, where group members are working on the latest cutting-edge technologies.

Priyanka Sharma is currently working as Vice President Projects - AI at Samyak Infotech Pvt Limited, a software-based company that has made its mark for providing end-to-end Enterprise Solutions for Logistics Management Companies in US, UK and several other hardware based projects like Stability Chambers and IoT based preventive maintenance systems for various Industries, Healthcare and Pharma sectors.

Ravi Sharma is working as a Professor in the Centre for Inter-Disciplinary Research and Innovation, University of Petroleum and Energy Studies, Dehradun, India. Dr Sharma is passionate in the field of Business analytics and worked in various MNC's as a leader of various software development groups. Dr Sharma has contributed various articles in the area of business analytics, prototype building for startup, and artificial intelligence. Dr Sharma leading academic institutions as a consultant to uplift research activities in inter-disciplinary domains