Design of Optimal Deep Learning based Disease Diagnosis Model for Cloud Centric IoT Healthcare Environment

1C. Nithyeshwari, 2Dr. G. Karthikeyan

1Department of Computer Science, Periyar Arts College, Cuddalore-1.
2Assistant Professor, Department of Computer Science, Periyar Arts College, Cuddalore-1.

Mail id: varshnicy2012@gmail.com, gkarthikeyan2007@gmail.com

Abstract

Recently, Cloud centric Internet of Things (IoT) technologies have been commonly employed in healthcare sector, which enables the seamless monitoring of patients’ health conditions. For effective medical diagnosis using the data collected by IoT devices, automated machine learning (ML) and deep learning (DL) models find useful. In this view, this study introduces an optimal deep learning enabled disease diagnosis model for cloud centric IoT (ODLDD-CCIoT) for healthcare environment. The proposed ODLDD-CCIoT technique comprises different stages of operations such as data acquisition, pre-processing, classification, and hyperparameter optimization. In addition, the ODLDD-CCIoT technique involves the design of convolution neural network-long short term memory (CNN-LSTM) model for disease diagnosis and classification. Moreover, the efficient teaching and learning based optimization (TLBO) technique can be utilized for the hyperparameter tuning of the CNN-LSTM model. A comprehensive experimental analysis is performed on heart disease (HD) and activity recognition datasets and the results are investigated in terms of different measures.

Keywords: Internet of things, Cloud computing, Deep learning, Machine learning, Healthcare diagnosis, Hyperparameter optimization, CNN-LSTM.

1. Introduction

IoT is an evolving trend for the next generation of technology that is referred to as interconnection of exclusively identified devices and smart objects. IoT is enclosed by several objects that are embedded invisibly in the environmental surroundings [1-3]. Furthermore, IoT provides appropriate solutions to massive applications like industry management, emergency service, health care, and traffic congestion. For efficiently monitoring remote patient health, IoT based medical application has gradually received significant attention [4]. With the continuous growth in Information and Communication Technology (ICT), medicinal sensor provides a resolution to several healthcare applications such as diagnosing chronic diseases,
providing elder healthcare, and patient activity monitoring at remote side. Furthermore, the accessibility of healthcare IoT devices results in a superior manner for diagnosing disease. As well, the usage of medical devices, diagnostic devices, etc. could be considered as smart devices developing a healthcare IoT platform [5].

IoT clubbed with cloud computing has become an effective platform to monitor patients at remote locations offering continuous healthcare data to caretakers and doctors. Cloud storage provides large number of storage and processing abilities in a scalable form [6]. Advancement in Cloud computing could manage parallel processing, resource sharing, security problems, and data service incorporation with scalable data storage. Furthermore, in the current scenarios, monitoring based cloud centric frameworks could be utilized to develop applications helpful in smart environments [7]. Over the past few years, the development of sensor technology and IoT associated with wearable healthcare devices have enhanced the patients’ health quality by using smart remote healthcare observing system. Currently, the cloud enabled IoT platform is extensively utilized in healthcare monitoring system and smart remote healthcare systems [8]. The integration IoT of and cloud has several advantages from resource management aspects like powerful processing, resource distribution, supporting user mobility in monitoring systems, and prevent from data fragmentation over different datasets [9]. A present remote healthcare monitoring scheme in cloud enabled IoT environments comprising a context where the patient’s information is stored and in cloud transmitted, and distributed with the aim of mining data from anytime and anywhere [10].

This study designs an optimal deep learning based disease diagnosis model for cloud centric IoT (ODLDD-CCIoT) for healthcare environment. The proposed ODLDD-CCIoT technique comprises different stages of operations such as data acquisition, pre-processing, classification, and hyperparameter optimization. In addition, the ODLDD-CCIoT technique involves the design of convolution neural network-long short term memory (CNN-LSTM) model for disease diagnosis and classification. Moreover, the efficient teaching and learning based optimization (TLBO) technique is utilized for the hyperparameter tuning of the CNN-LSTM model. A comprehensive experimental analysis is performed on two benchmark datasets and the results are investigated in terms of different measures.

2. Related works

This section offers a comprehensive review of existing disease diagnosis models for cloud enabled IoT environments. In [11], proposed a healthcare monitoring system for cloud based
IoT platform, where the patient’s health condition can be derived by forecasting diseases through mining information gathered from IoT gadgets and other healthcare reports. An efficient disease diagnoses method is utilized for analyzing the patient’s healthcare information with the purpose of providing a composite medical or health prescription. Afterward validating the results by healthcare team, can be forwarded to the patients. Next, the patients indicate her non-functional needs like time, location, and cost to discover a suitable composite medical or health services according to her preference. In [12], a remote healthcare monitoring system employs a lightweight block encryption technique to provide privacy for medical and healthcare information in cloud enabled IoT platforms are introduced. Here, the patient’s healthcare conditions are described by forecasting serious conditions via data mining approaches to analyze the genetic information collected through IoT device and encryptin technique is utilized for ensuring the patient confidential information turn out to be secured. They have a critical effect on this kind of system because of the limited resource in IoT environments.

In [13], a novel systematic method is utilized for the diabetes disease, and the correlated healthcare information is made on the basis of UCI datasets and the healthcare sensor for forecasting the patient infected with diabetes seriously. Further, they proposed a novel classifier termed Fuzzy Rule based Neural Classifiers to diagnose the severity and the disease. [14] proposed an optimum SVM to classify lung images where the parameter of SVM is enhanced and FS has been performed by the adapted GWO method integrated with genetic algorithm (GWO-GA). [15] proposed an EEPSOC method for the efficient selective of CHs amongst different IoT gadgets. They are employed to sense medical information is gathered in a form of cluster and a CH would be selected with the help of EEPSOC. The selected CH would transmit the information to the cloud. Next, the CH is accountable for forwarding information of the IoT device to the cloud. Then, ANN classifier is employed for diagnosing the medical information in the cloud to recognize the seriousness of the disease.

[16] presents a smart Skin Monitoring Device concept which enables patients in rural regions to remotely monitoring skin diseases. The presented approach comprises cloud and AI based IoT, in which CNN is employed for analyzing the disease predictions and healthcare images. [17] introduces a Tri-logical IoT fog cloud (TIFC) method for collecting AES information for controlling, and monitoring the data through the Spatio temporal method. Diverse activities are related to the Spatio temporal pattern through a time-sequence granule at a distinct timestamp. The FCM classifiers are utilized for analyzing the class of a patients-
based health correlated data parameter. Hence, for efficient health-related decision making and data delivery to the users, a predictive method depends on spatio temporal is utilized for managing the healthcare resource.

3. The Proposed Model

In this work, a novel ODLDD-CCIoT technique is designed for effective healthcare diagnosis in IoT enabled cloud environment. The proposed ODLDD-CCIoT model encompasses different subprocesses namely data acquisition, pre-processing, CNN-LSTM based classification, and hyperparameter optimization. Fig. 1 illustrates the workflow diagram of ODLDD-CCIoT technique.

![Workflow of ODLDD-CCIoT model](image)

3.1. Data Acquisition

The patient’s health record has been collected by implementing a data collection system that uses a continual involvement of intelligent medical gadgets. They can be located over the patient body to monitor the health status. At this point, the sensors are enclosed with wearable

www.turkjphysiotherrehabil.org
and inbuilt ones. Every sensor has been related to biosensors such as ECG, EEG, Blood Pressure (BP), etc. The sensor node has been appropriate for gathering structured as well as unstructured kinds of data. For retaining the data integrity, but implementing the broadcast tasks, the channel has been secured with use of Secure Socket Layer (SSL) for providing privacy. The timestamp synchronization of various types of sensors is applied.

3.2. Data Pre-processing

For providing effectual apply with minimum cost to data mining procedures, the quality of data has been optimum. The values missing from the database have been filled under the entire CKD dataset. In any case, if continuous features exist, the techniques are synchronized for building discrete traits. It can contain any noisy and missing values from all instances. For improving the performance of medicinal data, the input data has been pre-processed.

3.3. Design of CNN-LSTM based Classification Model

During classification, the CNN-LSTM model receives the preprocessed data as input to classify the existence of different class labels. The presented one dimensional CNN- LSTM method is comprised of input, 4 convolution layers (CL), 1 pooling layer (PL), 2 LSTM layers, softmax (SM), and 4 fully connected layers (FCL). Initially, the input data is straightforwardly utilized as the input of the presented method with the dimension of $178 \times 1$. Next, they are passed to the initial CL for extracting abstractive features of the input data, whereas the amount of one dimensional convolution kernel (CK) in the CL is 64, the dimension of every CK is $3 \times 1$ and the stride of convolution kernel is 1. The CL and ReLU activation layer could present nonlinearity to the presented method. Now, the arithmetical expression of the ReLU activation and the one dimensional convolution operation is determined in the following:

$$y^l_a = \sigma\left( \sum_{i=1}^{N_{l-1}} conv1D(w_{i,j}^l, x_{i}^{la-1}) + b^l_a \right)$$

In which $x_{i}^{la-1}$ denotes the $ith$ feature map (FM) in the $(l-1)th$ layer; $y^l_j$ signifies the $jth$ FM; $w_{i,j}^l$ signifies the trainable convolution kernel; $N_{l-1}$ symbolizes the amount of FM in the $(1-1)th$ layer; conv1D is the one dimensional convolutional function with no padding of 0’s; thus, the size of FM is lesser when compared to $la$- $1th$ layer; $b^l_j$ indicates the bias of $jth$ feature map in the $lth$ layer; $\sigma()$ indicates the ReLU function that could assist prevent the overfitting problems. It can be expressed by:
\[
\sigma(x) = \begin{cases} 
0, & x \leq 0 \\
 x, & x > 0 
\end{cases}
\]

(2)

Afterward, the activation and convolution, 64 FM using the size of 176 \times 1 [18]. Then, the outcome of the CL1 is later passed by a max pooling (MP) layer. Now, the arithmetical expression of the one dimensional MP function is defined in the following:

\[
p_i^a = \max \left( p_i^{a'} : a \leq a' < a + s \right)
\]

(3)

Whereas \( p_i^{a'} \) represent the \( a'th \) neuron in \( ith \) FM beforehand MP operation and \( p_i^a \) indicates the \( a \)th neuron in \( ith \) FM afterward MP operation, and \( s \) signifies the size of pooling window. In Pooling Layerl, the stride of windows is also 2 and the size of pooling window is 2. It could considerably decrease the amount of trained variables in the presented method and speed up the trained model. Afterward, in the pooling operation, 64 FMs using the size of 88 \times 1 are outputted. Next, 3 CL are followed for additionally extracting high level features that could facilitate the classification. It consists of CL4, CL3, CL2, there exist 128 kernels in the shape of 3 \times 1 in the CL 2, 512 kernels in a similar shape in the CL3, and 1024 kernels in a similar shape in the CL4. Likewise, the convolutional operation is similar to the CL1, and also ReLU is employed for nonlinear activation.

Afterward, the FMs passed by each one dimensional CL, the achieved 1024 FMs with the size of 82 \times 1 would be fed to 1 FCL using 256 neurons and then dropout is employed to the outcome of the FCL. FCL can integrate the output from the CL and decrease the dimensions of FM for fitting the input of LSTM layer, and dropout could lessen the over-fitting problems to certain range. After passed by the FCLI, the output feature is fed to the LSTM layer i.e., able to avoid the long term dependencies problems in the traditional RNNs. It consists of forgetting, state, output, and input gates in the LSTM cell. It integrates with one another for preserving the prior data and additionally enhance the capacity of learning beneficial healthcare data. There exit 64 neurons in the LSTM Layer 1 and the LSTM Layer 2. Next, the feature passed by the LSTM layer, the final feature would be fed to 3 FCLs. FCL2, FCL3, and FCL4 are FCLs with 256, 128, and 64 neurons, correspondingly. Lastly, SM layer is added to the presented method. The complete structure of the presented method is altered based on certain epileptic seizure identification tasks.
3.4. Process involved in TLBO based Hyperparameter Optimization

At the final stage, the hyperparameters involved in the CNN-LSTM model can be optimally chosen by the use of TLBO algorithm. TLBO, i.e., derived from the conventional method of learning in classroom, consists of 2 major segments. The optimization method of TLBO begins with an initialization of population known as students. Afterward the initial estimation of solution, the optimal solutions defines the teacher. The information is distributed among the teachers and another solution in the teacher stage. The TLBO method stages are formulated and presented in the following:

Phase I (generating the primary population): In TLBO method, the first candidate solution is regarded as a class using $nS$ student. The collection of randomized students ($S$) are generated by:

$$ S = Lb + (Ub - Lb) \times rand(nS, nVar) $$

Whereas $nS$ represent the student count, $nVar$ denotes the amount of design parameters, and $Lb & Ub$ indicate the minimum and maximum bounds of the design parameters.

Phase II (teacher stage): In the beginning, the students are calculated, and their respective penalized objective function vector ($PFit$) is produced. Then, the optimal student is selected as the teacher ($T$). A step size upgrades the student towards the teacher, which can be achieved on the basis of the knowledge of the teacher and the average knowledge of each student ($AveS$). It is represented in the following equation:

$$ stepsize_i = T - TF_i \times AveS $$

$$ newS = S + rand_{i,j} \times stepsize $$

$$ i = 1, 2, \ldots, nS \text{ and } j = 1, 2, \ldots, nVar $$

The word of $srepsize_i$ is the step size of $ith$ student, $newS$ indicates the vector of novel student, $rand^{is}$ an arbitrary value selected in the range of zero and one as well as the teacher factor ($TF_i$) is consider to adjust the effects of the teacher knowledge on the class average, that is 1 or 2. The value of $TF_i$ isn’t provided as an input to, and the arbitrarily determines its value. It can be useful for exploring a wide area of the novel solution by adapting an approach, like hybridization or another enhancement method stated in QTLBO algorithm.
Phase III (replacement approach): In this phase, recently created student is calculated and substituted by their consistent old ones in a greedy method. In this manner, the recently created students with a better penalized objective function are selected to his equivalent old one. Thus, a novel class using $nS$ student is generated.

Phase IV (learner stage): In this stage, initially, all the students arbitrarily selects another student ($S_{rs}$) excepting himself [19]. Then, the student shares his knowledge with the arbitrarily elected one. The student move towards another elected student when another elected one has additional knowledge than him ($PFit_i < PFit_{rs}$). The learner stage is expressed by:

$$\text{stepsize}_i = \begin{cases} S_j - S_{rs}; & PFit_i < PFit_{rs} \\ S_{rs} - S_j; & PFit_i \geq PFit_{rs} \end{cases}$$

$$\text{newS} = S + \text{rand}_{i,j} \times \text{stepsize}$$

$$i = 1, 2, ..., nS \text{ and } j = 1, 2, ..., nVar$$

Phase V (replacement approach): The replacement approach is replicated.

Phase VI (End condition): When the end condition is fulfilled, the process is ended. Or else, go to phase 2.

The TLBO algorithm derives a fitness function for the optimal parameter tuning of the CNN-LSTM model. For tuning the hyperparameters, ten-fold cross validation process is performed and classification accuracy is considered as the fitness function of the TLBO algorithm. It can be defined as follows:

$$\text{Fitness Function (Average Accuracy)} = \frac{\Sigma_{n=1}^{N} \text{TestAccuracy}(n)}{N}$$

where average accuracy in fitness function denotes the testing, accuracy offered by the CNN-LSTM model with ten fold cross validation process.

4. Experimental Validation

This section investigates the performance analysis of the ODLDD-CCIoT model in terms of different measures. The results are examined on benchmark HAR and HD dataset. The HD dataset holds a set of 303 samples with 76 features collected from a set of 14 patients [20]. In addition, the HAR dataset [21] includes a collection of 30 persons doing 6 activities with static and dynamic ways. The dataset holds 70% of training rate and 30% of testing data.
4.1. Results analysis on HD Dataset

This section investigates the performance of the ODLDD-CCIoT model on the HD dataset. The confusion matrix generated by the ODLDD-CCIoT model on the classification of HD is shown in Fig. 2. The figure reported that the ODLDD-CCIoT model has classified a total of 145 instances into absence of HD and 132 instances into presence of HD.

![Confusion Matrix](image1)

**Fig. 2.** Confusion matrix of ODLDD-CCIoT model on HD

Fig. 3 demonstrates the ROC analysis of the ODLDD-CCIoT model on the HD dataset. The results portrayed that the ODLDD-CCIoT model has resulted in an increased ROC of 0.97.

![ROC Curve](image2)

**Fig. 3.** ROC analysis of ODLDD-CCIoT model on HD
Fig. 4. Accuracy analysis of ODLDD-CCIoT model under HD

Fig. 4 demonstrates the training and validation accuracy analysis of the ODLDD-CCIoT model on HD dataset. The figure exhibited that the validation and training accuracy are consistently increased with a rise in epoch count. It is also observed that the validation accuracy seems to be higher than the training accuracy.

The training and validation loss analysis of the ODLDD-CCIoT model on the test HD dataset is depicted in Fig. 5. The results ensured that the ODLDD-CCIoT model has resulted in minimal training and validation loss. Particularly, the validation loss is found to be considerably lower than the training loss.

Table 1 and Fig. 6 showcases the classification results analysis of the ODLDD-CCIoT with CNN-LSTM models on the test HD dataset. The CNN-LSTM model has gained effective
outcome with the accuracy, precision, recall F1-score, AUC, kappa, and MCC of 92.260%, 92.180%, 92.390%, 92.230%, 92.390%, 84.480%, and 84.570% respectively. In addition, the ODLDD-CCIoT model has reached effectual outcomes with the accuracy, precision, recall F1-score, AUC, kappa, and MCC of 93.270%, 93.230%, 93.490%, 93.250%, 93.490%, 86.520%, and 86.720% correspondingly.

Table 1 Classification results analysis ODLDD-CCIoT Model on HD Dataset

<table>
<thead>
<tr>
<th>Measures</th>
<th>CNN-LSTM</th>
<th>ODLDD-CCIoT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy Score</td>
<td>92.260</td>
<td>93.270</td>
</tr>
<tr>
<td>Precision</td>
<td>92.180</td>
<td>93.230</td>
</tr>
<tr>
<td>Recall</td>
<td>92.390</td>
<td>93.490</td>
</tr>
<tr>
<td>F1-Score</td>
<td>92.230</td>
<td>93.250</td>
</tr>
<tr>
<td>AUC Score</td>
<td>92.390</td>
<td>93.490</td>
</tr>
<tr>
<td>Kappa Score</td>
<td>84.480</td>
<td>86.520</td>
</tr>
<tr>
<td>MCC</td>
<td>84.570</td>
<td>86.720</td>
</tr>
</tbody>
</table>

Fig. 6. Result analysis of ODLDD-CCIoT model under HD dataset

In order to showcase the improved performance of the ODLDD-CCIoT technique, a brief comparative accuracy analysis is made in Table 2 and Fig. 7. The figure reported that the NN-
Fuzzy, DT, and NN0GA models have resulted in reduced accuracy of 0.8, 0.8068, and 0.8099 respectively. Along with that, the DT-GR, ELM, and SVM models have accomplished moderately closer accuracy of 0.841, 0.865, and 0.8676 respectively.

**Table 2** Results analysis of Proposed ODLDD-CCIoT with Existing Models on HD Dataset

<table>
<thead>
<tr>
<th>Methods</th>
<th>Accuracy Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ODLDD-CCIoT</td>
<td>0.9327</td>
</tr>
<tr>
<td>VNB-LR</td>
<td>0.8741</td>
</tr>
<tr>
<td>NN-Fuzzy</td>
<td>0.8000</td>
</tr>
<tr>
<td>Decision Tree</td>
<td>0.8068</td>
</tr>
<tr>
<td>ELM Model</td>
<td>0.8650</td>
</tr>
<tr>
<td>SVM</td>
<td>0.8676</td>
</tr>
<tr>
<td>NN-GA</td>
<td>0.8099</td>
</tr>
<tr>
<td>DT-GR</td>
<td>0.8410</td>
</tr>
</tbody>
</table>

**Fig. 7.** Accuracy analysis of ODLDD-CCIoT model under HD dataset
Moreover, the VNB-LR technique has tried to attain near optimal accuracy of 0.8741. However, the ODLDD-CCIoT technique has accomplished better performance with a higher accuracy of 0.9327.

4.2. Results analysis on HAR dataset

This section investigates the performance of the CNN-LSTM model on the HAR dataset. The confusion matrix generated by the CNN-LSTM methodology on the classification of HAR is shown in Fig. 8. The figure outperformed that the CNN-LSTM technique has ‘walking’ classified a total of 145 instances into HAR. Similarly, the figure reported that the CNN-LSTM method has ‘walking_upstairs’ classified a total of 1538 instances into HAR. Also, the figure reported that the CNN-LSTM model has ‘walking_downstairs’ classified a total of 1402 instances into HAR. Besides, the figure stated that the CNN-LSTM model has ‘sitting’ classified a total of 1623 instances into HAR. In addition, the figure reported that the CNN-LSTM model has ‘Standing’ classified a total of 1938 instances into HAR.

At the same time, the figure stated that the CNN-LSTM technique has ‘laying’ classified a total of 1958 instances into HAR. Followed by, the figure portrayed that the CNN-LSTM approach has ‘stand-to-sit’ classified a total of 69 instances into HAR. Likewise, the figure reported that the CNN-LSTM model has ‘sit-to-stand’ classified a total of 30 instances into HAR. Along with that, the figure reported that the CNN-LSTM model has ‘sit-to-stand’ classified a total of 95 instances into HAR. Then, the figure described that the CNN-LSTM model has ‘lie-to-sit’ classified a total of 0 instances into HAR. Moreover, the figure reported that the CNN-LSTM model has ‘stand-to-lie’ classified a total of 104 instances into HAR. Eventually, the figure stated that the CNN-LSTM methodology has ‘lie-to-stand’ classified a total of 81 instances into HAR.
Fig. 8. Confusion Matrix of HAR-Optimal CNN-LSTM Model

Fig. 9 demonstrates the ROC analysis of the CNN-LSTM model on the HAR dataset. The results portrayed that the CNN-LSTM model has resulted in an increased ROC of 1.00.

Fig. 9. ROC of HAR-Optimal CNN-LSTM Model
Fig. 10. Accuracy analysis of CNN-LSTM model under HAR dataset

Fig. 10 exhibits the training and validation accuracy analysis of the CNN-LSTM approach on HAR dataset. The figure demonstrated that the validation and training accuracy are consistently enhanced with a rise in epoch count. It can be also observed that the validation accuracy seems to be superior to the training accuracy.

Fig. 11. Loss analysis of CNN-LSTM model under HAR dataset

The training and validation loss analysis of the CNN-LSTM technique on the test HAR dataset is depicted in Fig. 11. The outcomes ensured that the CNN-LSTM method has resulted to lower training and validation loss. Particularly, the validation loss is found to be considerably lesser than the training loss.
Table 3 and Fig. 12 depict the classification outcomes analysis of the ODLDD-CCIoT with CNN-LSTM algorithms on the test HAR dataset. The CNN-LSTM model has reached effectual outcome with the accuracy, precision, recall F1-score, AUC, kappa, and MCC of 94.380%, 61.100%, 70.080%, 64.310%, 84.780%, 93.350%, and 93.400% respectively. Furthermore, the ODLDD-CCIoT model has gained effective results with the accuracy, precision, recall F1-score, AUC, kappa, and MCC of 96.620%, 82.670%, 86.400%, 83.730%, 93.040%, 96.010%, and 96.030% correspondingly.

**Table 3 Results analysis of Proposed ODLDD-CCIoT Model on HAR Dataset**

<table>
<thead>
<tr>
<th>Measures</th>
<th>CNN-LSTM</th>
<th>ODLDD-CCIoT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy Score</td>
<td>94.380</td>
<td>96.620</td>
</tr>
<tr>
<td>Precision</td>
<td>61.100</td>
<td>82.670</td>
</tr>
<tr>
<td>Recall</td>
<td>70.080</td>
<td>86.400</td>
</tr>
<tr>
<td>F1-Score</td>
<td>64.310</td>
<td>83.730</td>
</tr>
<tr>
<td>AUC Score</td>
<td>84.780</td>
<td>93.040</td>
</tr>
<tr>
<td>Kappa Score</td>
<td>93.350</td>
<td>96.010</td>
</tr>
<tr>
<td>MCC</td>
<td>93.400</td>
<td>96.030</td>
</tr>
</tbody>
</table>

**Fig. 12. Result analysis of ODLDD-CCIoT model under HAR dataset**

For demonstrating the improved performance of the ODLDD-CCIoT approach, a brief comparative accuracy analysis is made in Table 4 and Fig. 13 [22-24]. The figure reported that the TDD, TSN, and P-CNN models have resulted in lower accuracy of 0.6590, 0.6940, and
0.8760 respectively. Also, the 2-stage HMM, CNN, and MLCNN models have accomplished to moderately closer accuracy of 0.9318, 0.9375, and 0.9479 respectively. In addition, the TFFT-MLCNN technique has tried to attain near optimum accuracy of 0.9575. At last, the ODLDD-CCIoT technique has accomplished better performance with a superior accuracy of 0.9662.

**Table 4** Results analysis of Proposed ODLDD-CCIoT with Existing Models on HAR Dataset

<table>
<thead>
<tr>
<th>Measures</th>
<th>Accuracy Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>ODLDD-CCIoT</td>
<td>0.9662</td>
</tr>
<tr>
<td>CNN</td>
<td>0.9375</td>
</tr>
<tr>
<td>P-CNN</td>
<td>0.8760</td>
</tr>
<tr>
<td>TSN</td>
<td>0.6940</td>
</tr>
<tr>
<td>TDD</td>
<td>0.6590</td>
</tr>
<tr>
<td>2-Stage HMM</td>
<td>0.9318</td>
</tr>
<tr>
<td>MLCNN</td>
<td>0.9479</td>
</tr>
<tr>
<td>TFFT-MLCNN</td>
<td>0.9575</td>
</tr>
</tbody>
</table>

*Fig. 13. Accuracy analysis of ODLDD-CCIoT model under HAR dataset*
5. Conclusion

In this paper, a novel ODLDLL-CCIoT technique is derived for the effective diagnosis of diseases in the IoT enabled cloud environment. The ODLDCCIoT technique has the ability to properly identify the existence of the diseases with maximum detection rate. Besides, the ODLDCCIoT technique involves diverse sub processes namely data acquisition, pre-processing, CNN-LSTM based classification, and hyperparameter optimization. Moreover, the application of the TLBO algorithm to tune the parameters of the CNN-LSTM model results in improved classification performance. The performance analysis of the ODLDCCIoT technique takes place using two benchmark datasets namely HD and HAR dataset. A detailed comparison study highlighted the improved outcomes of the ODLDCCIoT technique. As a part of future scope, the ODLDCCIoT technique can be enhanced by the use of image processing techniques for medical imaging diagnosis.

References


