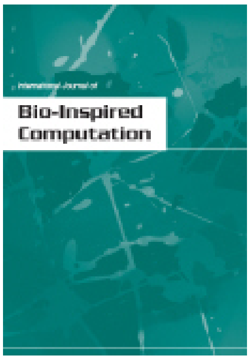


International Journal of Bio-Inspired Computation

ISSN online: 1758-0374 - ISSN print: 1758-0366
https://www.inderscience.com/ijbic

Deep recurrent neural network-based Hadoop framework for COVID prediction with applications to big data in cloud computing

D.B. Jagannadha Rao, Vijayakumar Polepally, S. Nagendra Prabhu, Parsi Kalpana

DOI: 10.1504/IJBIC.2023.10055199

Article History:

Received:	03 September 2021
Last revised:	09 June 2022
Accepted:	19 June 2022
Published online:	04 April 2023

Deep recurrent neural network-based Hadoop framework for COVID prediction with applications to big data in cloud computing

D.B. Jagannadha Rao*

Department of Computer Science and Engineering (Data Science),
Malla Reddy University, Hyderabad,
Telangana, India
Email: dbjagannadharao007@gmail.com
*Corresponding author

Vijayakumar Polepally

Department of Computer Science and Engineering,
Kakatiya Institute of Technology and Science,
Warangal, Telangana, India
Email: vijayakumarpolepally@gmail.com

S. Nagendra Prabhu

Department of Computational Intelligence,
School of Computing,
SRM Institute of Science and Technology,
Kattankulathur, Chennai, Tamilnadu, India
Email: snagendraprabhu@gmail.com

Parsi Kalpana

Department of Computer Science and Engineering,
Vasavi College of Engineering Hyderabad,
Telangana, India
Email: prem.kalpana@gmail.com

Abstract: This paper proposes a particle squirrel search optimisation-based deep recurrent neural network (PSSO-based DRNN) to predict the coronavirus epidemic (COVID). Here, the cloud-based Hadoop framework is used to perform the prediction process by involving the mapper and reducer phases. Initially, the technical indicators are extracted from the time series data. Then, the deep belief network (DBN) is employed for feature selection from the technical indicators. After that, the COVID prediction is done by the DRNN classifier trained using the PSSO algorithm. The PSSO is developed by the integration of particle swam optimisation (PSO) and squirrel search algorithm (SSA). The PSSO-based DRNN is compared with existing methods and obtained minimal MSE and RMSE of 0.0523, and 0.2287 by considering affected cases. By considering death cases, the proposed method achieved minimal MSE and RMSE of 0.0010, and 0.0323 and measured minimum MSE of 0.0049 and minimum RMSE of 0.0702 for recovered cases.

Keywords: COVID-19; MapReduce; cloud; deep belief network; DBN; deep recurrent neural network.

Reference to this paper should be made as follows: Rao, D.B.J., Polepally, V., Prabhu, S.N. and Kalpana, P. (2023) 'Deep recurrent neural network-based Hadoop framework for COVID prediction with applications to big data in cloud computing', *Int. J. Bio-Inspired Computation*, Vol. 21, No. 1, pp.36–47.

Biographical notes: D.B. Jagannadha Rao is working as an Associate Professor in Department of Data Science (CSE) at Malla Reddy University, Hyderabad, Telangana, India. He received his MTech degree in 2008 from Jawaharlal Nehru Technological University, Hyderabad, India. He received his PhD degree from Shri Jagdishprasad Jhabarmal Tibrewala University (JJTU). Jhunjhunu, Rajasthan, India in 2021. He has more than 11 years of academic experience. He published 13 research papers in national and international journals and conferences. His teaching and research interests include algorithm design, cloud computing, and data science. He is a life member of IAENG.

Vijayakumar Polepally is working as an Associate Professor in the Department of Computer Science and Engineering at Kakatiya Institute of Technology and Science, Warangal, Telangana, India. He received his PhD degree from JNTU Hyderabad in 2020. He has four years of Software Industry experience and ten years of Academic experience. He published 15 research papers in national and international journals and conferences. His teaching and research interests include algorithm design, cloud computing, data science, machine learning and artificial intelligence. He is a life member of ISTE, India.

S. Nagendra Prabhu is currently working as an Assistant Professor, Department of Computational Intelligence, School of Computing, SRM Institute of Science and Technology, Kattankulathur, Chennai, India. He completed his PhD in Network Security in Cloud Computing from Anna University, Chennai, India. He has total teaching and research experience of 12+ years. His research interest includes cloud computing, Botnet attack, web based network security. Currently, the author is doing research related security issues in machine learning. He published papers in more than 25 international journals and national peer reviewed journals. He has delivered many guest lectures across colleges in India.

Parsi Kalpana is currently working as an Associate Professor in Department of Computer Science and Engineering at Vasavi College of Engineering, Ibrahimbagh, Hyderabad. She received her PhD in Computer Science Engineering from Osmania University and holds two Master's, MTech in Computer Science and Engineering from JNTUH and Master of Computer Applications from OU. She possesses more than 19 years of teaching experience, and she has published 16 papers in various international journals and conferences. Her area of interest is cloud computing and its security mechanisms. She is a life member of IAENG, India.

1 Introduction

With the increasing and continuous growth of various information dissemination techniques, the data in the cloud computing and internet of things (IoT) technologies (Kumar and Vimala, 2020). Thorat (2021) is constantly increasing. Also, the global information scale is continuously increasing at the rate of two times for every two years (Wu et al., 2014). The development of big data (Zhang et al., 2021) shows many challenges and complexities other than understandable advantages (Kuang et al., 2014; Cai et al., 2021). Big data is commonly used in different fields, like business (Kumar et al., 2019), medicine, and industry Raisi (2020). A major issue in handling big data is to design algorithms for classification (Ambati and Gayar, 2021) and pattern recognise (Ulfarsson et al., 2016). The popular framework in big data (Srinivas et al., 2019) follows the method called Apache Hadoop, which offers the distributed system transparently through the implementation of MapReduce framework (Dean and Ghemawat, 2008) distribute data between the nodes, while MapReduce is allowed for data processing by reducing the data transfer Rathod (2020) or exploiting the data locality through the network by executing the local data (Nimmani et al., 2021; Wilder, 2012).

An innovative solution is required to analyse, manage and develop big data (Elkano et al., 2017) in the increasing network on patient details, subjects that are infected, and community migration incorporated with the public health data (Bojja et al., 2021; Bojja and Ambati, 2020), genomics, pharmaceutical, and clinical trials (Zhao et al., 2020). Various data source comprises web articles, online communications, text messages, and social media Singh (2021) are widely helpful to analyse the infection growth for community behaviour. To integrate the data with artificial intelligence (AI) and machine learning (ML), the researchers need to forecast when and where the disease is spread such that the regions can be identified for matching it with the needed arrangements (Ambati et al., 2020). The infrastructure required to store and analyse the big data for further processing is a cost-effective and efficient manner. However, this process can be organised using AI solutions and cloud computing (Tuli et al., 2020a). In Huang et al. (2020), the cloud solutions (Netaji and Bhole, 2020) are designed to fight against the coronavirus and to find the peak outbreak considered to achieve 98% of accuracy in the real world test cases in china. Various pneumonia categories are resolved by the ML-based CT image analysis solution that is effectively considered to examine the COVID-19 patients (Depeursinge et al., 2015). The designing of vaccines for COVID-19 Gupta et al. (2020) is focused on the analysis of molecular docking and genome sequences using different AI and ML methods (Jin et al., 2020; Tuli et al., 2020b).

The main contribution of the research is provided as follows:

- Proposed PSSO-based DRNN: A prediction scheme is designed to predict the pandemic disease COVID using a DRNN classifier. The technical indicators are extracted and are enabled to select the features using DBN. To predict COVID cases, DRNN classifier is adopted using the MapReduce framework, where the process of technical feature extraction is carried out in the mapper and the feature selection and prediction mechanism is carried out in the reducer phase.

The remaining part of the paper is ordered as follows: The review of different prediction methods is described in Section 2, and Section 3 describes the proposed model. The cloud-based Hadoop framework is discussed in Section 4 presents, Section 5 discusses about results and discussion, and the paper end in Section 6.

2 Literature survey

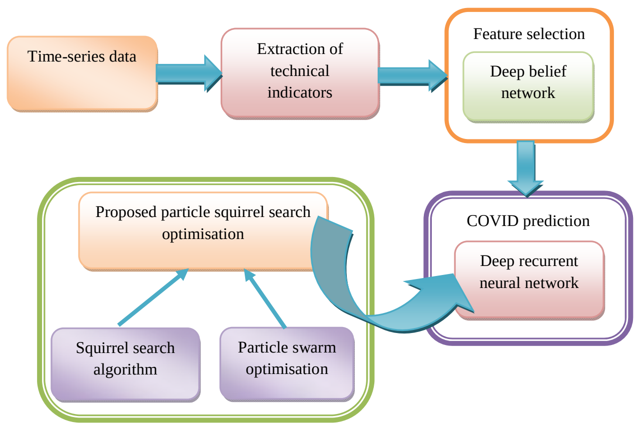
This section discusses the review of different prediction methods with their advantages and disadvantages. Rustam et al., (2020) introduced an ML-based prediction model to forecast infected COVID cases. It used the dataset that contains actual past data to predict future days. It helps to decide at the time but failed to use the appropriate and accurate forecasting methods. Hasan (2020) introduced a hybrid model based on artificial neural network (ANN) and ensemble empirical mode decomposition (EEMD) for COVID-19 pandemic prediction. This method was an effective indicator for predicting COVID. It failed to perform multivariate analysis for time-series data. Vennila and Kannan (2019) introduced a parallel linguistic fuzzy rule along with canopy MapReduce (LFR-CM). It reduced the consumptional time. It increased the runtime of classification with big data using the MapReduce framework. Giuliani et al. (2020) introduced a Spatio – temporal model to predict the COVID cases. This method offered better forecasts by finding the number of infected cases at the local level and reduced the delay reporting. However, this model hs high computational complexity.

Tuli et al. (2020a) introduced an ML-based prediction scheme for predicting the potential threat of COVID worldwide. This method obtained a better fit by iterative weighting the prediction model. It was designed in the cloud model to generate real-time and accurate predictions by predicting the growth behaviour of an epidemic. It failed to take a correct decision that affects the situation of public health. Chen et al. (2016) introduced a parallel random forest (PRF) model for data classification in the cloud. Here, the vertical data partitioning model was used to minimise the communication cost and the data multiplexing model was utilised to diminish or reduce the volume of data. The dimension reduction model was used to increase the accuracy of the algorithm. However, it is further required to increase the task scheduling and data allocation model in the distributed environment. Waheed et al. (2020) introduced an auxiliary classifier generative adversarial network (ACGAN) for predicting the global pandemic. Here, CNN was used for classifying the data into two classes, namely COVID or normal. It increased the accuracy of detection by generating synthetic images. However, it failed to enhance the quality of synthetic images by training GAN. Jindal et al. (2018) developed a fuzzy rule-based classifier to decide on data classification. Here, the membership function was used to infer the data for performing the process of fuzzification and defuzzification. This method was highly effective in identifying the patients who suffered from the diseases. The computational cost of this approach was high.

3 Proposed particle squirrel search optimisation-based deep recurrent neural network for COVID prediction in cloud-based Hadoop framework

To predict the pandemic disease more accurately and to enhance the prediction performance result a challenging task in recent decades. Therefore, the newly designed algorithm named PSSO-based DRNN is adopted for COVID cases prediction more accurately. The cloud-based Hadoop framework is designed to perform the prediction process. The technical features like average directional movement index (ADMI), simple moving average (SMA), average true range (ATR), rate of change (ROCR), Williams %R, exponential moving average (EMA), and Stochastic %K are extracted from technical indicators extraction phase. The DBN is used for feature selection process at the mapper phase and the selected features are employed to predict the COVID cases using DRNN classifier. The training of DRNN is done using the proposed PSSO algorithm, which is the integration of SSA (Jain et al., 2019) and PSO (Wang et al., 2018), respectively. Figure 1 portrays the schematic view of the proposed method for COVID prediction.

Figure 1 Schematic view of proposed PSSO-based DRNN for COVID prediction (see online version for colours)



3.1 Data representation

Time-series is the real-valued sequence that specifies the values of the variable for time. Every record here belongs to the time label and its corresponding cases, like recovered, affected, and death cases of COVID data. Consider the time-series data as d with n number of time samples is given as,

$$D = \{d_i\}; 1 \le i \le n \tag{1}$$

3.2 Extraction of technical indicators

The technical indicators, such as SMA, EMA, Williams %R, ROCR, ATR, ADMI, and Stochastic %K (Gandhmal and Kumar, 2020; Kelotra and Pandey, 2020; Zhang et al., 2018) are extracted from the time series data.

The technical indicators obtained from time-series COVID data is specified as,

$$T = \{T_{ij}\}; 1 \le i \le n, 1 \le j \le 7 \tag{2}$$

Here, T specifies the technical features acquired from COVID data.

3.3 Feature selection by DBN

The technical features acquired from the time-series data are denoted as T, which is enabled to select the unique and the optimal features to achieve the process of COVID prediction. Here, the deep learning model named DBN (Zou et al., 2015) is employed for optimal feature selection by concerning layer-wise weights w and w' on technical features. Here, layer-wise weights are specified as the reconstruction weights that are considered to compute the reconstruction error for the technical features. The feature variations generally formulate the dissimilarity of the reconstruction error. The features with less reconstruction error can be more reconstructible.

Let $T = \{T_{ij}\}$ be the technical features and T'_{ij} as the reconstructed features that correspond to T_{ij} and hence the reconstruction error of the technical features T_{ij} is computed as, $E_{ij} \|T'_{ij} - T_{ij}\|$ and the average error is computed as,

$$\hat{E} = \frac{1}{n*7} \sum_{i=1}^{n} \sum_{j=1}^{7} E_{ij} \tag{3}$$

At each iteration of the learning model, the threshold σ is computed based on β. Here, the parameter β specifies the ratio of outliers to technical features. The feature outliers can be removed at the iterative learning process such that the reconstruction weight matrix is highly reliable to achieve feature reconstruction. The procedure applied to get the weight matrix from the training data to find the unique features for the testing process is described here. Let P be the reconstruction weight matrix, T be the technical features, and f_{ik} specifies the total number of features acquired from T, then reconstructed features F is computed as,

$$F = \delta(f_{ik}, P) \tag{4}$$

Here, δ denotes reconstruct factor.

The reconstruction error E_{re} with respect to f_{ik} is given as,

$$E_{re} = \|F - f_{ik}\| \tag{5}$$

The features with less reconstruction error are selected as discriminative features. Hence, it is required to filter the features having larger reconstruction errors. To select features, the parameter η is specified as,

$$\eta = (1 - \beta)^{\hbar} \tag{6}$$

Here, β denotes the parameter defining the outliers ratio, $\hbar$ specifies the real number of iterations. The error values are sorted to find the error value of the order η. The selected features T is represented as,

$$F^{selected} = \{f_{il}\}; 1 \le i \le n, 1 \le l \le k \tag{7}$$

Here, F selected denotes the features taken from the technical indicators using DBN.

3.4 COVID prediction using proposed PSSO-based DRNN

After unique and optimal features, selection, the COVID prediction is performed with DRNN classifier. The training of DRNN is done by the proposed PSSO algorithm.

3.4.1 Structure of DRNN

DRNN (Ma et al., 2019) is more efficient and effective in processing the technical features of time-series data, as it has a strong relationship among current data samples with that of the previous one. The reason for using DRNN classifier for COVID prediction is that this classifier can use to model the time-series data more effectively. Hence, each data sample is considered to be dependent on the previous one.

Let $F^{selected}$ be the features selected, the previously hidden layer is termed as b_{v-1}, and output found using hidden layer is specified as g_v. The hidden state at v^{th} layer is given as,

$$b_v = x\left(SF^{selected} + Rb_{v-1} + c\right) \tag{8}$$

where, S specifies the weight matrix between input and hidden state, the weight matrix from current to next state is indicated as R, bias vector is specified as c, and $x(.)$ denotes non-linear activation function. The output state is determined as,

$$g_v = q\left(Ab_v + K\right) \tag{9}$$

where, q denotes non-linear activation function, A represents the weight matrix between hidden and output state, and K signifies bias vector.

3.4.2 Training procedure of DRNN using PSSO algorithm

The DRNN is trained with proposed optimisation algorithm named PSSO algorithm, which is the integration of PSO (Wang et al., 2018) and SSA (Jain et al., 2019), respectively. The algorithmic steps in the proposed PSSObased DRNN is explained as follows,

1 *Population initialisation*: In the forest G, let us assume h number of flying squirrels such the initial position of flying squirrel is modelled as,

$$G_r = G_{z+}B(0,1) \times \left(G_p - G_z\right) \tag{10}$$

where, G_z specifies lower bound of r^{th} flying squirrel in μ^{th} dimension, $B(0,1)$ denotes random number uniformly distributed such that it lies in the range of 0 to 1, G_r specifies the position of flying squirrel, and upper bound is specified as G_p of r^{th} flying squirrel in μ^{th} dimension.

2 *Compute fitness measure*: It is evaluated to find the optimal solution and the function to compute fitness measure is represented as,

$$H = \frac{1}{\varepsilon}\sum_{\upsilon=1}^{\varepsilon}\left[g\upsilon - O\upsilon\right] \tag{11}$$

where, H denotes fitness. O_υ denotes the actual optimum value for function under optimisation and ε terms the amount of samples. The output state computation is mentioned as g_v.

3 *Sort the location of G*: The location of squirrel G is sorted in ascending order for fitness measure. Accordingly, the squirrel that has minimal fitness value is moved to a hickory nut tree and the foraging behaviour of squirrels gets affected because of the presence of predators. The squirrels sort their location with the probability of predator γ.

4 *Generate a new solution*: Based on the foraging behaviour of squirrels, the following situations are defined. The squirrel in the forest moves to search for food when there is no predator. They move to a hiding location when there is a predator and hence, the foraging behaviour of the squirrels is designed with three cases.

- *Case-1*: Here, the squirrels in the acorn nut trees G_L move to hickory nut tree, and hence the new position formed in this scenario is represented as,

$$G_L(y+1) = G_L(y) + Z * \chi \times \left(G_J(y) - G_L(y)\right) \tag{12}$$

where, Z indicates gliding distance, y signifies current iteration, χ represents gliding constant which is set as 1.9 and $G_J(y)$ indicates the location of a squirrel that moved to hickory nut tree. Moreover, the gliding distance of flying squirrel is specified as,

$$Z = \frac{Y}{\tan\phi} \tag{13}$$

Here, Y indicates weight loss.

- *Case-2:* The squirrels in the normal trees G_H move to acorn nut trees to manage their daily energy needs and is represented using the below equation as,

$$G_H(y+1) = G_H(y) + Z * \chi \times \left(G_L(y) - G_H(y)\right) \tag{14}$$

$$G_H(y+1) = G_H(y) + Z\chi G_L(y) - Z\chi G_H(y) \tag{15}$$

$$G_H(y+1) = G_H(y)[1 - Z\chi] + Z\chi G_L(y) \tag{16}$$

The standard equation of PSO is represented as,

$$G_H(y+1) = G_H(y) + U_H(y+1) \tag{17}$$

$$\begin{aligned} G_H(y+1) = G_H(y) &+ BU_H(y) \\ &+ s_1 q_1 \left(A_H(y) - G_H(y)\right) \\ &+ s_2 q_2 \left(A_H(y) - G_H(y)\right) \end{aligned} \tag{18}$$

$$\begin{aligned} G_H(y+1) = G_H(y) &+ BU_H(y) + s_1 q_1 A_H(y) \\ &- s_1 q_1 G_H(y) + s_2 q_2 A_H(y) \\ &- s_2 q_2 G_H(y) \end{aligned} \tag{19}$$

$$\begin{aligned} G_H(y+1) = G_H(y)[1 - s_1 q_1 - s_2 q_2] &+ BU_H(y) \\ &+ s_1 q_1 A_H(y) - s_2 q_2 A(y) \end{aligned} \tag{20}$$

$$\begin{aligned} G_H(y)[1 - s_1 q_1 - s_2 q_2] = G_H(y+1) &+ BU_H(y) \\ &+ s_1 q_1 A_H(y) - s_2 q_2 A(y) \end{aligned} \tag{21}$$

$$G_H(y) = \frac{G_H(y+1) + BU_H(y) + s_1 q_1 A_H(y) - s_2 q_2 A(y)}{\left[1 - s_1 q_1 - s_2 q_2\right]} \tag{22}$$

By substituting the above equation (26) in equation (20) is represented as,

$$G_H(y+1) = \frac{\begin{array}{c} G_H(y+1) - BU_H(y) - s_1 q_1 A_H(y) \\ s_2 q_2 A_H(y) - s_2 q_2 A(y) \end{array}}{\left[1 - s_1 q_1 - s_2 q_2\right]} \tag{23}$$

$$\begin{aligned} &G_H(y+1) - \frac{G_H(y)[1 - Z\chi]}{\left[1 - s_1 q_1 - s_2 q_2\right]} = \\ &\frac{-BU_H(y) - s_1 q_1 A_H(y) - s_2 q_2 A(y)}{\left[1 - s_1 q_1 - s_2 q_2\right]}[1 - Z\chi] \\ &+ Z\chi G_L(y) \end{aligned} \tag{24}$$

$$\begin{aligned} &G_H(y+1) \\ &= \frac{1 - s_1 q_1 - s_2 q_2}{Z\chi - s_1 q_1 - s_2 q_2}\left[\begin{array}{l} \dfrac{-BU_H(y) - s_1 q_1 A_H(y) - s_2 q_2 A(y)}{\left[1 - s_1 q_1 - s_2 q_2\right]} \\ \left[1 - Z\chi\right] + Z\chi G_L(y) \end{array}\right] \end{aligned} \tag{25}$$

where, y signifies current iteration, χ represents gliding constant, and $U_H(y)$ denotes the velocity of the particle.

- *Case-3:* The squirrels in the normal trees that already consumed acorn nuts try to move to the hickory nut tree to store hickory nuts to consume at the time of scarcity for food.

$$G_H(y+1) = G_H(y) + Z * \chi \times \left(G_L(y) - G_H(y)\right) \tag{26}$$

5 *Compute seasonal constant V*: For the flying squirrel, the seasonal constant is computed using the below equation as,

$$V^y = \sqrt{\sum_{\lambda=1}^{M}\left(G_L, \lambda - G_J, \lambda\right)^2} \tag{27}$$

where, y specifies current iteration. Here, the minimal value of the seasonal constant is computed as,

$$V_{\min} = \frac{10e^{-6}}{(365)^{y/(y_\varpi/2.5)}} \tag{28}$$

where, y and y_ϖ specifies current and denotes maximum iteration.

6 *Random relocation of a flying squirrel*: The flying squirrels cannot search for the hickory nuts tree to find the food source. Here, the squirrels relocation is computed as,

$$G_H^{new} = G_z + \delta(\kappa) + \left(G_p - G_z\right) \tag{29}$$

$$\delta(\kappa) = 0.01 \times \frac{N_1 \times \omega}{|N_2|^{1/W}} \tag{30}$$

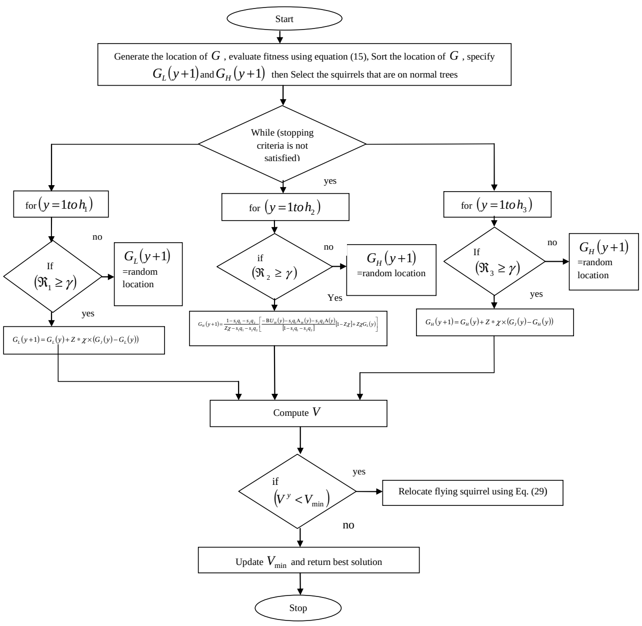
$$\omega = \left[\frac{T(1+W) \times \sin\left(\frac{\pi W}{2}\right)}{T\left(\frac{1+W}{2}\right) \times W \times 2^{\left(\frac{W-1}{2}\right)}} \right]^{1/W} \tag{31}$$

$$T(\vartheta) = (\vartheta - 1)1! \tag{32}$$

where, δ indicates levy distribution that considered to enhance the of global exploration capability. Here, N_1 and N_2 are random number distributed randomly with the range 0 and 1 and W specifies parameter in constant with value 1.5, respectively.

7 *Termination*: The steps above are repeated until it obtain the best solution. Figure 2 shows the flowchart for the proposed PSSO-based DRNN.

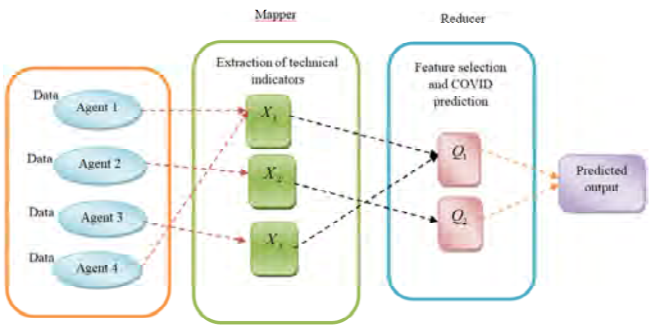
Figure 2 Flowchart for the proposed PSSO-based DRNN



4 Cloud-based Hadoop version 3.1.4 framework for COVID prediction

The cloud environment is designed with the data owner and cloud server by connecting several cloud agents using the Map Reduce framework. The agents may be located anywhere in the network and these agents are connected with the data owner to share their data. The agent collects the data from the local region and stores the data through the data owner, which captures the data from several agents and stores the data in the cloud for further processes. Figure 3 represents the Map Reduce framework for COVID prediction. The Map Reduce framework is designed with two phases, namely the mapper phase and reducer phase that helps to achieve COVID prediction using time-series data. In the mapper phase, the technical indicators are extracted and in the reducer phase, the feature selection process and the prediction process are carried out to generate the predicted result. Let us specify the number of agents in the network as agent 1, agent 2, agent 3, and so on. The mappers present in the mapper phase is mentioned as X_m. The reducers present in the reducer phase are denoted as Q_u that selects the features. Moreover, the COVID prediction process is performed in the reducer phase, where the prediction results is generated using the DRNN classifier. The structure of the MapReduce framework is designed with four agents, three mappers, and two reducers, respectively. The data captured by the first agent is processed by X_1mapper, and the data acquired by agent 2 is processed by X_2mapper, the data captured by agent 3 is processed by X_3 and the data of agent 4 is processed by X_1mapper. As, only three mappers are available the data incoming from agent 4 is fed to X_1 in a round-robin fashion. Similarly, the result of X_1mapper is given to Q_1, X_2 to Q_2 and the output of X_3 is fed to Q_1 in a round-robin fashion. Finally, the predicted output is generated from the reducer phase in MapReduce framework using the DRNN classifier.

Figure 3 Mapreduce framework of COVID prediction (see online version for colours)



5 Results and discussion

This section describes the results obtained by the proposed PSSO-based DRNN classifier for COVID prediction by considering the narrow and global data.

5.1 *Experimental setup*

The implementation of the proposed method is performed in the MATLAB tool using the dataset specified in (https://data.humdata.org/dataset/novel-coronavirus-2019-ncov-cases).

- COVID-19 cases data: The data of this dataset are compiled from different sources, including DXY.cn, BNO news, the world health organisation (WHO), the national health commission of the people's republic of

China (NHC), and so on. It maintains the 2019 novel COVID-19 data repository. The parameter setup for the proposed approach is mentioned in the Table 1.

Table 1 Parameter setup

Parameters	*Values*
Number of hidden units	200
Layers	sequence InputLayer(numFeatures), lstmLayer(numHiddenUnits), regressionLayer
Options	Training Options – 'adam'
Maximum Epochs	200
Gradient threshold	1
Initial learning rate	0.005
Learning rate schedule	piecewise
Learning rate drop period	125
Learning rate drop factor	0.2
Verbose 1	0.2

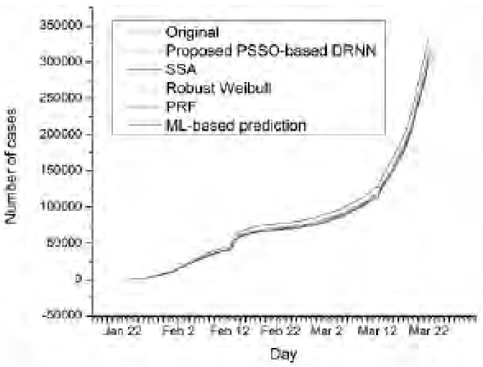
5.3 Comparative methods

The performance is measured using the existing methods, like ML-based prediction (Rustam et al., 2020), parallel random forest (PRF) (Chen et al., 2016), Robust Weibull (Tuli et al., 2020b), and SSA (Jain et al., 2019).

5.4 Comparative analysis

Here, the analysis is carried out to find the prediction result with global and narrow databased on affected, death, and recovered cases.

Figure 4 Prediction result of affected cases with global data (see online version for colours)



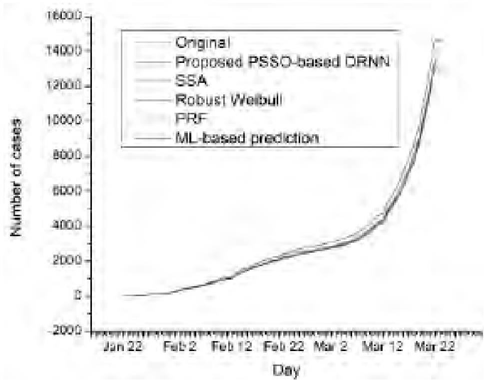
5.4.1 Analysis of affected cases with global data

Figure 4 shows the prediction result obtained for affected cases with global data. On February 12, the original affected cases is 45221, whereas the affected prediction outcome using existing SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN is 40699, 41152, 40699, 40247, and 42056 in such a way that the error acquired between the original affected cases for predicted cases using SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN is 4522, 4069, 4522, 4974, and 3165. Similarly, the original affected cases on March 22 is 335955, and the affected prediction outcome obtained using the proposed PSSO-based DRNN is 312439, which shows better performance when compared with the existing method.

5.4.2 Analysis of death cases with global data

The prediction result obtained for death cases by considering global data is portrayed in Figure 5. On February 2, the original death cases is 362, whereas the death prediction outcome using existing SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN is 326, 30, 326, 323, and 337 in such a way that the error acquired between the original death cases with respect to predicted death cases using SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN is 36, 32, 36, 39, and 25. Likewise original death cases occur on March is also predicted as provided in the graph, in which the error value measured between original and the predicted death cases using the proposed PSSO-based DRNN is 377 and the existing SSA, Robust Weibull, PRF, and ML-based prediction is 424, 472, 472, 566, and 377.

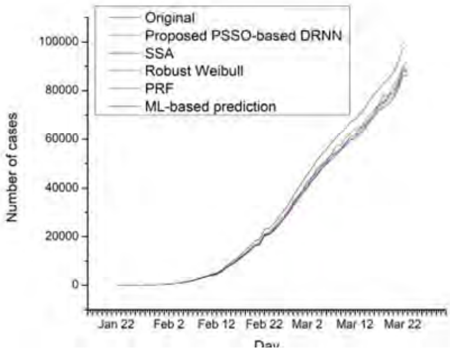
Figure 5 Prediction result of death cases with global data (see online version for colours)



5.4.3 Analysis of recovered cases with global data

Figure 6 represents the prediction result obtained for recovered cases with global data. The original cases recovered on February 2 is 472, whereas the predicted outcome obtained for the by considering the recovered cases using existing SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN is 425, 425, 421, 421, and 435 in such a way that the error acquired between the original recovered cases for predicted recovered cases using SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN is 47, 47, 51, 51, and 37. In such a way the error acquired between the original recovered cases for predicted recovered cases on March is also analysed.

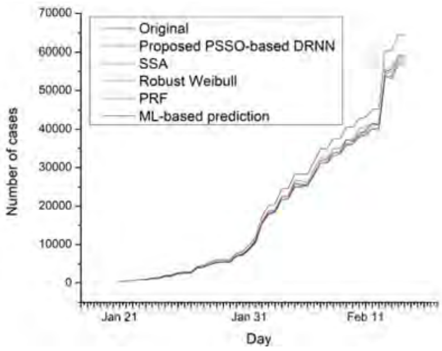
Figure 6 Prediction result of recovered cases with global data (see online version for colours)



5.4.4 *Analysis of affected cases with narrow data*

Figure 7 shows the prediction result obtained for affected cases with narrow data. The original affected cases observed on January 21 is 330, whereas the prediction outcome achieved for the affected cases using existing SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN is 304, 297, 297, 291, and 311 and hence the error value computed between the original and predicted affected cases using SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN is 26, 33, 33, 39, and 19. At February 11, the original affected cases is 45117, whereas the affected prediction outcome for the proposed method is 41508 in such a way that the difference between the original affected cases and prediction outcome of the proposed method is 3609, which shows improved result compared to the existing methods.

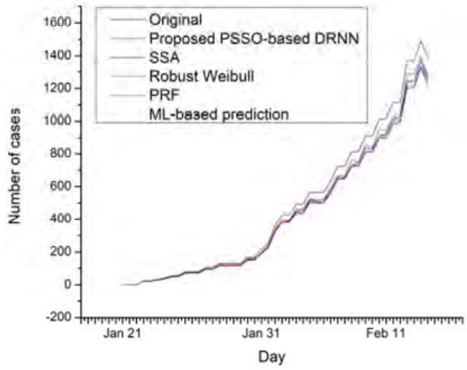
Figure 7 Prediction result of affected cases with narrow data (see online version for colours)



5.4.5 *Analysis of death cases with narrow data*

The prediction result obtained for death cases by considering narrow data is portrayed in Figure 8. In January 31, the original death cases is 213, whereas the death prediction outcome using existing SSA, Robust Weibull, PRF, and ML-based prediction, is 196, 94, 190, 190, and the proposed PSSO-based DRNN 196. In such a way that the error acquired between the original death cases with respect to predicted death cases in February 11 is also discussed.

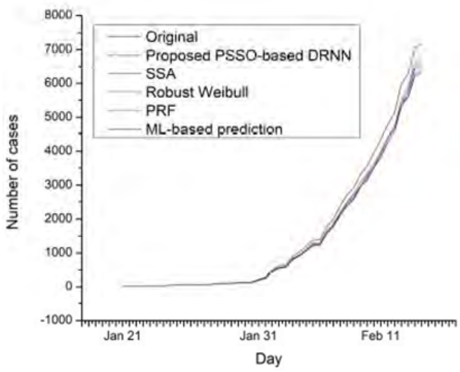
Figure 8 Prediction result of death cases with narrow data (see online version for colours)



5.4.6 *Analysis of recovered cases with narrow data*

Figure 9 represents the prediction result obtained for recovered cases with narrow data. The original cases recovered on January 11 is 220, whereas the recovered cases predicted using existing SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN is 201, 201, 198, 194, and 203 such that the error obtained between the original and the predicted recovered cases for the proposed PSSO-based DRNN is 17.

Figure 9 Prediction result of recovered cases with narrow data (see online version for colours)

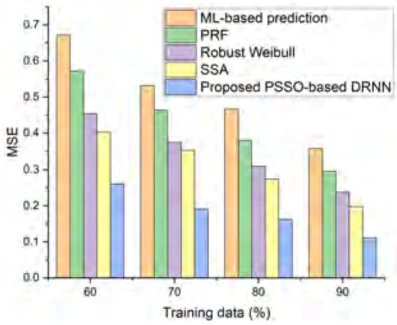


5.4.7 *Analysis with global data*

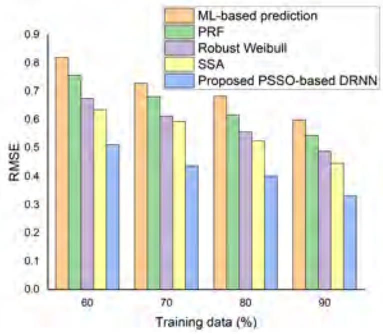
The analysis of the proposed PSSO-based DRNN made using global data by considering affected cases is shown in Figure 10. The comparative analysis of MSE with affected cases is portrayed in Figure 10(a). When the training data as 80%, the MSE obtained by SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN is 0.4659, 0.3795, 0.3091, 0.2740, and 0.1618. The MSE observed SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN at 90% training data is 0.3568, 0.2951, 0.2376, 0.1977, and 0.1103.

Figure 10(b) depicts the analysis of RMSE by considering affected cases. At 90% training data, the RMSE obtained by SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN is 0.5973, 0.5433, 0.4875, 0.4446, and 0.3321.

Figure 10 Analysis of proposed PSSO-based DRNN with global data by considering affected cases (a) MSE (b) RMSE (see online version for colours)



(a)

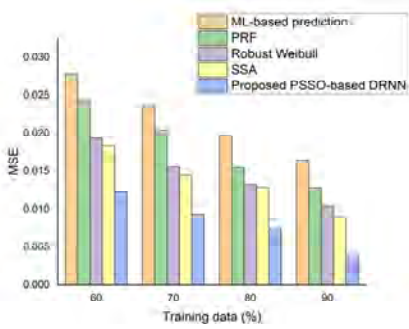


(b)

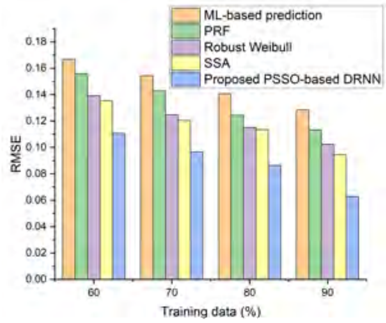
Figure 11 depicts the analysis of the proposed PSSObased DRNN evaluated using global data by considering death cases. Figure 11(a) shows the analysis of MSE in terms of death cases. The MSE achieved by SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN is at 80% training data is 0.0198, 0.0155, 0.0132, 0.0129, and 0.0074. The analysis made using RMSE by considering death cases is represented in Figure 11(b). The RMSE of SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN is 0.1284, 0.1133, 0.1023, 0.0944, and 0.0628 at 90% training data.

The analysis proposed PSSO-based DRNN made using global data by considering recovered cases is shown in Figure 12. Figure 12(a) represents the analysis of MSE based on the recovered cases. By considering the training data as 80%, the MSE of SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN is 0.1841, 0.1496, 0.1199, 0.1107, and 0.0579. The analysis for RMSE by considering recovered cases is represented in Figure 12(b). The RMSE computed by considering 80% training data of SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN is 0.4290, 0.3868, 0.3463, 0.3328, and 0.2407.

Figure 11 Analysis of proposed PSSO-based DRNN with global data by considering death cases (a) MSE (b) RMSE (see online version for colours)

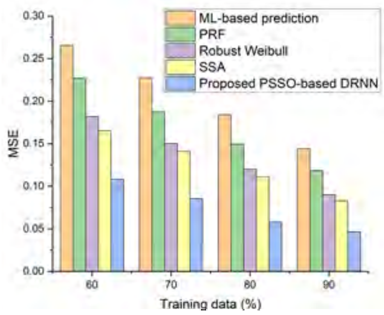


(a)

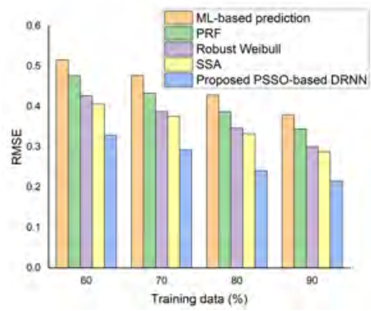


(b)

Figure 12 Analysis of proposed PSSO-based DRNN with global data by considering recovered cases (a) MSE (b) RMSE (see online version for colours)



(a)

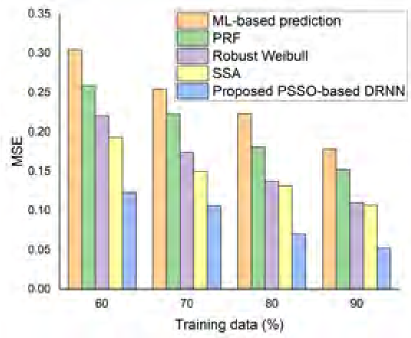


(b)

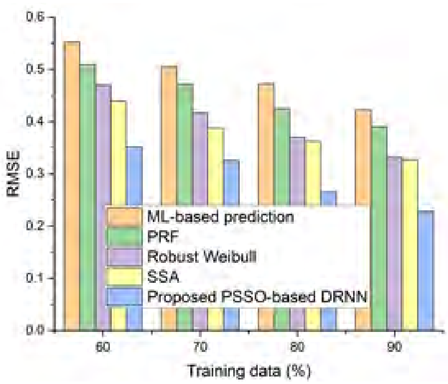
5.4.8 Analysis with narrow data

Figure 13 reveals the analysis of the proposed PSSO-based DRNN made using narrow data by considering affected cases. Figure 13(a) portrays the analysis of MSE based on the affected cases. For 80% training data, the MSE of SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN is 0.2230, 0.1808, 0.1372, 0.1313, and 0.0703. The analysis of RMSE was carried out using proposed PSSO-based DRNN by considering affected cases are shown in Figure 13(b). By considering 80% training data the RMSE of SSA, Robust Weibull, PRF, and ML-based prediction is 0.4722, 0.4252, 0.3704, 0.3624, and proposed PSSO-based DRNN is 0.2652.

Figure 13 Analysis of proposed PSSO-based DRNN with narrow data by considering affected cases (a) MSE (b) RMSE (see online version for colours)



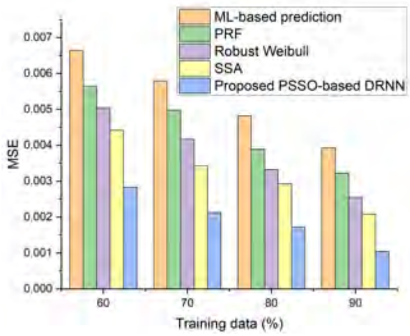
(a)



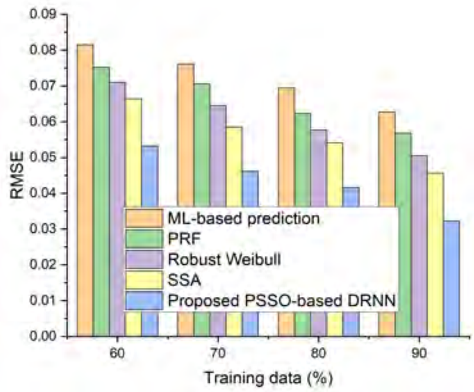
(b)

Figure 14 show the analysis of the proposed PSSO-based DRNN evaluated using narrow data by considering death cases. Figure 14(a) shows the analysis of MSE in terms of training data by considering death cases. At 80% training data, the MSE computed by SSA, Robust Weibull, PRF, and ML-based prediction is 0.0048, 0.0039, 0.0033, and 0.0029, whereas the proposed PSSO-based DRNN achieved a lower error value of 0.0017 and next to the proposed method ML-based prediction method shows lowest MSE value and SSA shows highest MSE value. The analysis carried out in terms of RMSE based on death cases is represented in Figure 14(b). The RMSE of SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN at 80% training data is 0.0694, 0.0624, 0.0576, 0.0541, and 0.0415.

Figure 14 Analysis of proposed PSSO-based DRNN with narrow data by considering death cases (a) MSE (b) RMSE (see online version for colours)



(a)

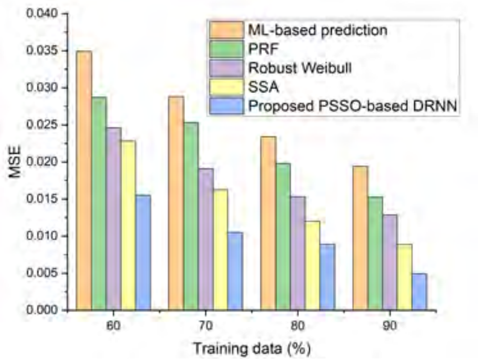


(b)

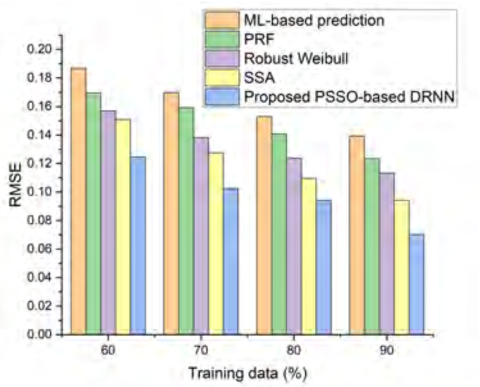
The analysis proposed PSSO-based DRNN made using narrow data by considering recovered cases is shown in Figure 15. Figure 15(a) portrays the analysis of MSE based on recovered cases with narrow data. The MSE of SSA, Robust Weibull, PRF, and ML-based prediction at 80% training data is 0.0234, 0.0198, 0.0153, and 0.0120, whereas for proposed PSSO-based DRNN achieved an MSE of 0.0089. The analysis of RMSE computed by considering recovered cases is represented in Figure 16(b). The RMSE of SSA, Robust Weibull, PRF, and ML-based prediction, and proposed PSSO-based DRNN at 90% training data is 0.1393, 0.1234, 0.1133, 0.0942, and 0.0702.

When compared to the existing methods, the proposed method has high performance. The reasons for the high performance of the proposed method are listed here. DBN in the proposed method is used for selecting unique features from the extracted technical indicators. DBN has the benefits like it is effective for vanishing gradient problems and they are very accurate and needs a small labelled dataset and select the unique features effectively. Next to feature selection cloud-based Hadoop framework is used to perform the prediction process using mapper and reducer phases, which has the advantages like fault tolerant, high throughput, performance, low network traffic, etc. Thus prediction is performed effectively. The features are then processed by the DRNN classifier to find the presence of COVID, which has the advantages like it can process input of any length and it does not increase model the size even if the input size is larger. The training of DRNN is done with proposed PSSO algorithm, has the advantages like improved convergence speed and easy implementation.

Figure 15 Analysis of proposed PSSO-based DRNN with narrow data by considering recovered cases (a) MSE (b) RMSE (see online version for colours)



(a)



(b)

6 Conclusions

In this research, the process of COVID prediction is achieved using proposed PSSO-based DRNN classifier. The proposed method performs the prediction process by involving the phases, like extraction of technical indicators, feature selection, and COVID prediction. The proposed method has the advantages of easy implementation, simple concept, computational efficiency, and robustness to control parameters, and obtained minimal MSE and RMSE of 0.0523 and 0.2287 by considering affected cases. By considering death cases, the proposed method achieved minimal MSE and RMSE of 0.0010, and 0.0323 and measured minimum MSE of 0.0049 and minimum RMSE of 0.0702 for recovered cases. The future dimension of research would be the improvement of prediction performance using some other deep learning classifier.

References

Ambati, L.S. and Gayar, O.E. (2021) 'Human activity recognition: a comparison of machine learning approaches', *Journal of the Midwest Association for Information Systems*, Vol. 2021, No. 1, pp.49–60.

Ambati, L.S., Gayar, O.F.E. and Nawar, N. (2020) 'Influence of the digital divide and socio-economic factors on prevalence of diabetes', *Issues in Information Systems*, Vol. 21, No. 4, pp.103–113.

Bojja, G.R. and Ambati, L.S. (2020) 'A novel framework for crop pests and disease identification using social media and AI', *MWAIS.*

Bojja, G.R., Liu, J. and Ambati, L.S. (2021) 'Health Information systems capabilities and Hospital performance–An SEM analysis', *AMCIS, 2021*.

Cai, X., Cao, Y., Ren, Y., Cui, Z. and Zhang, W. (2021) 'Multi-objective evolutionary 3D face reconstruction based on improved encoder–decoder network', *Information Sciences*, Vol. 581, pp.233–248.

Chen, J., Li, K., Tang, Z., Bilal, K., Yu, S., Weng, C. and Li, K. (2016) 'A parallel random forest algorithm for big data in a spark cloud computing environment', *IEEE Transactions on Parallel and Distributed Systems*, Vol. 28, No. 4, pp.919–933.

Dean, J., Ghemawat, S. (2008) 'Mapreduce: Simplified data processing on large clusters', *Commun. ACM*, Vol. 51, No. 1, pp.107–113.

Depeursinge, A., Chin, A.S., Leung, A.N., Terrone, D., Bristow, M., Rosen, G and Rubin, D.L. (2015) 'Automated classification of usual interstitial pneumonia using regional volumetric texture analysis in high-resolution CT', *Investigative Radiology*, Vol. 50, No. 4, p.261.

Elkano, M., Galar, M., Sanz, J. and Bustince, H. (2017) 'CHI- BD: a fuzzy rule-based classification system for big data classification problems', *Fuzzy Sets and Systems*, Vol. 348, pp.75–101.

Gandhmal, D.P. and Kumar, K. (2020) 'Wrapper-Enabled feature selection and CPLM-Based NARX model for stock market prediction', *The Computer Journal*, Vol. 64, No. 2, pp.169–184.

Giuliani, D., Dickson, M.M., Espa, G. and Santi, F. (2020) 'Modelling and predicting the spatio-temporal spread of Coronavirus disease 2019 (COVID-19) in Italy', *BMC Infectious Diseases*, Vol. 20, No. 700, pp.1–10.

Gupta, S., Raghuwanshi, G.S. and Chanda, A. (2020) 'Effect of weather on COVID-19 spread in the US: a prediction model for India in 2020', *Science of The Total Environment*, Vol. 728.

Hasan, N. (2020) 'A methodological Approach for predicting COVID-19 epidemic using EEMD-ANN hybrid model', *Internet of Things*, Vol. 11, p.100228.

Huang, C., Wang, Y., Li, X., Ren, L., Zhao, J., Hu, Y., Zhang, L., Fan, G., Xu, J. and Gu, X. (2020) 'Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China', *The Lancet*, Vol. 395, No. 10223, pp.497–506.

Jain, M., Singh, V. and Rani, A. (2019) 'A novel nature-inspired algorithm for optimization: Squirrel search algorithm', *Swarm and Evolutionary Computation*, Vol. 44, pp.148–175.

Jin, S., Wang, B., Xu, H., Luo, C., Wei, L., Zhao, W., Hou, X., Ma, W., Xu, Z. and Zheng, Z. (2020) 'Ai-assisted CT imaging analysis for covid-19 screening: building and deploying a medical AI system in four weeks', *MEDRXIV*.

Jindal, A., Dua, A., Kumar, N., Das, A.K., Vasilakos, A.V. and Rodrigues, J.J. (2018) 'Providing healthcare-as-a- service using fuzzy rule based big data analytics in cloud computing', *IEEE Journal of Biomedical and Health Informatics*, Vol. 22, No. 5, pp.1605–1618.

Kelotra, A. and Pandey, P. (2020) 'Stock market prediction using optimized deep-convlstm model', *Big Data*, Vol. 8, No. 1, pp.5–24.

Kuang, L., Hao, F. and Y. L.T. (2014)'A tensor-based approach for big data representation and dimensionality reduction', *Emerging Topics in Computing, IEEE Transactions on*, Vol. 2, No. 3, pp. 280–291.

Kumar, C.A. and Vimala, R. (2020) 'Load balancing in cloud environment exploiting hybridization of chicken swarm and enhanced raven roosting optimization algorithm', *Multimedia Research*, Vol. 3, No. 1, pp.45–55.

Kumar, P.V., Srichandana, H. and Srinivas, C. (2019) 'Rating prediction from social Users', *International Journal of Research*, Vol. 8, No. 5, pp.1703–1719.

Ma, A., Filippi, A.M., Wang, Z. and Yin, Z. (2019) 'Hyperspectral image classification using similarity measurements-based deep recurrent neural networks', *Remote Sensing*, Vol. 11, No. 2, p.194.

Netaji, V.K., Bhole, G.P. (2020) 'Optimal container resource allocation using hybrid SA-MFO algorithm in cloud architecture', *Multimedia Research*, Vol. 3, No. 1, pp.11–20.

Nimmani, P., Vodithala, S and Polepally, V. (2021) 'Neural network based integrated model for information retrieval', in *Proceedings of 2021 5th International Conference on Intelligent Computing and Control Systems* (ICICCS), pp.1286–1289.

Novel Coronavirus (COVID-19) Cases Data: [online] https://data.humdata.org/dataset/novel-coronavirus-2019-ncov-cases (accessed June 2021).

Raisi, A.A.J.A. (2020) 'Hybird particle swarm optimization and gravitational search algorithm for economic dispatch in power system', *Journal of Computational Mechanics, Power System and Control*, Vol. 3, No. 3, pp.34–40.

Rathod, S. (2020) 'Hybrid metaheuristic algorithm for cluster head selection in WSN', *Journal of Networking and Communication Systems*, Vol. 3, No. 4, pp.1–8.

Rustam, F., Reshi, A.A., Mehmood, A., Ullah, S., On, B., Aslam, W. and Choi, G.S. (2020) 'COVID-19 future forecasting using supervised machine learning models', *IEEE Access*, Vol. 8, pp.101489–101499.

Singh, Y. (2021) 'Hybrid dragonfly and particle swarm optimization algorithm for congestion management', *Journal of Computational Mechanics, Power System and Control*, Vol. 4, No. 2, pp.10–17.

Srinivas, C., Sree, M.N.J. and Kumar, P.V. (2019) 'Database sharing between Hadoop Clusters using peer – to peer Dela sharing', *International Journal of Research*, Vol. 8, No. 5, pp.2094–2102.

Thorat, V.A. (2021) 'Cloud intrusion detection using modified crow search optimized based neural network', *Journal of Networking and Communication Systems*, Vol. 4, No. 2, pp.9–15.

Tuli, S., Tuli, S., Tuli, R. and Gill, S.S. (2020) 'Predicting the growth and trend of COVID-19 pandemic using machine learning and cloud computing', *Internet of Things*, Vol. 11, p.100222.

Tuli, S., Tuli, S., Wander, G., Wander, P., Gill, S.S., Dustdar, S., Sakellariou, R. and Rana, O. (2020) 'Next generation technologies for smart healthcare: challenges, vision, model, trends and future directions', *Internet Technology Letters*, Vol. 3, No. 2, p.e145.

Ulfarsson, M.O., Palsson, F., Sigurdsson, J and Sveinsson, J.R. (2016) 'Classification of big data with application to imaging genetics', *Proceedings of the IEEE*, Vol. 104, No. 11, pp.2137–2154.

Vennila, V. and Kannan, A.R. (2019) 'Hybrid parallel linguistic fuzzy rules with canopy mapreduce for big data classification in cloud', *International Journal of Fuzzy Systems*, Vol. 21, No. 3, pp.809–822.

Waheed, A., Goyal, M., Gupta, D., Khanna, A., Al-Turjman, F. and Pinheiro, P.R. (2020) 'Covidgan: data augmentation using auxiliary classifier gan for improved covid-19 detection', *IEEE Access*, Vol. 8, pp.91916–91923.

Wang, D., Tan, D. and Liu, L. (2018) 'Particle swarm optimization algorithm: an overview', *Soft Computing*', Vol. 22, No. 2, pp.387–408.

Wilder, B. (2012) *Cloud Architecture Patterns*, O'Reilly Media, Sebastopol, California.

Wu, X., Zhu, X. and Wu, G.Q. (2014) 'Data mining with big data', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 26, No. 1, pp.97–107.

Zhang, C. et al. (2020) 'A novel scoring system for prediction of disease severity in COVID-19', *Frontiers in Cellular and Infection Microviology*, Vol. 10, p.318.

Zhang, C., Ji, Z., Zhang, J., Wang, Y., Zhao, X. and Yang, Y. (2018) 'Predicting Chinese stock market price trend using machine learning approach', in *Proceedings of the 2nd International Conference on Computer Science and Application Engineering*, pp.1–5.

Zhang, Z., Cao, Y., Cui, Z., Zhang, W and Chen, J. (2021) 'A many-objective optimization based intelligent intrusion detection algorithm for enhancing security of vehicular networks in 6G', *IEEE Transactions on Vehicular Technology*, Vol. 70, No. 6, pp.5234–5243.

Zhao, S., Lin, Q., Ran, J., Musa, S.S., Yang, G., Wang, W., Lou, Y., Gao, D., Yang, L. and He, D. (2020) 'Preliminary estimation of the basic reproduction number of novel coronavirus (2019-ncov) in china, from 2019 to 2020: A data-driven analysis in the early phase of the outbreak', *International Journal of Infectious Diseases*, Vol. 92, pp.214–217.

Zou, Q., Ni, L., Zhang, T. and Wang, Q. (2015) 'Deep learning based feature selection for remote sensing scene classification', *IEEE Geoscience and Remote Sensing Letters*, Vol. 12, No. 11, pp.2321–2325.