INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 18, Issue: 1, Month: February, Year: 2023
Article Number: 4396, https://doi.org/10.15837/ijccc.2023.1.4396

CCC Publications

Covid-19 Forecasting Using CNN Approach With A Halbinomial Distribution And A Linear Decreasing Inertia Weight-Based Cat Swarm Optimization

Rajesh Khanna Murugesan, Karthikeyan Madhu, Jayalakshmi Sambandam, Lakshmanan Malliga

Rajesh Khanna Murugesan
Department of Information Technology
Vel Tech Multi Tech Dr. Rangarajan Dr. Sakunthala Engineering College, Avadi, Chennai-600116, India
Corresponding author: seenum84@gmail.com

Karthikeyan Madhu
Department of EEE
Vel Tech Multi Tech Dr. Rangarajan Dr. Sakunthala Engineering College, Avadi, Chennai-600116, India
karthickm37@gmail.com

Jayalakshmi Sambandam
MEASI Institute of Information Technology, Chennai, India
jayalakshmi.research@gmail.com

Lakshmanan Malliga
Malla Reddy Engineering College for Women (Autonomous), Maisammaguda Secunderabad-500100, India
dr.malligalakshmanan18@gmail.com

Abstract

In recent times, the COVID-19 epidemic has spread to over 170 nations. Authorities all around the world are feeling the strain of COVID-19 since the total of infected people is rising as well as they does not familiar to handle the problem. The majority of current research effort is thus being directed on the analysis of COVID-19 data within the framework of the machines learning method. Researchers looked the COVID 19 data to make predictions about who would be treated, who would die, and who would get infected in the future. This might prompt governments worldwide to develop strategies for protecting the health of the public. Previous systems rely on Long Short-Term Memory (LSTM) networks for predicting new instances of COVID-19. The LSTM network findings suggest that the pandemic might be over by June of 2020. However, LSTM may have an over-fitting issue, and it may fall short of expectations in terms of true positive. For this issue in COVID-19 forecasting, we suggest using two methods such as Cat Swarm Optimization (CSO) for reducing the inertia weight linearly and then artificial intelligence based binomial distribution is used. In this proposed study, we take the COVID-19 predicting database as an contribution and normalise it using the min-max approach. The accuracy of classification is improved with the use of the first method to choose the optimal features. In this method, inertia weight is added to the CSO optimization algorithm convergence. Death and confirmed cases are predicted for a certain time period throughout India using Convolutional Neural Network with Partial Binomial Distribution based on carefully chosen characteristics. The experimental findings validate that the suggested scheme performs better than the baseline system in terms of f-measure, recall, precision, and accuracy.

Keywords: Min-Max Normalization, Cat Swarm Optimization (CSO), Binomial Distribution, and COVID-19 forecasting.

1 Introduction

In December of 2019, Wuhan, China will be the epicentre of a global pandemic using a newly discovered coronavirus called COVID-19 [1-3]. An uptick in reported cases of COVID-19 has been seen in a majority of nations during December 2019. According to a report by the Globe World Health organisation issued May 29, 2020, there have been around 353 373 fatalities and 5,596,550 total cases registered over the world. COVID 19 illness is more common in the elderly and those with compromised immune systems or other medical conditions, particularly those affecting the lungs.

The symptoms of COVID-19 are similar to those of the flu, including difficulty breathing, a cold, and a cough [4]. So yet, neither COVID-19 nor SARS-CoV-2 have any approved vaccines or medications that are shown to be successful therapeutically. In the absence of pharmacological therapies, different nations have taken varied approaches to managing the COVID-19 epidemic. The widespread deployment of lockdown is the most typical tactic [5].

On January 30, 2020, a student returning from Wuhan, the expansive metropolis of Hubei province in China, was diagnosed with corona virus in the Thrissur district of Kerala [6]. The indian government has ordered a nationwide lockdown beginning March 25, 2020, and lasting for 21 days, including a day of "Janata Curfew" on March 22, 2020, in an effort to stop the spread of the corona virus, also known as SARS-CoV-2.

The shutdown has been extended by the Indian government because of the widespread spread of the corona virus. Washing one's hands often, refraining from touching one's mouth, nose, or ears, and maintaining an emotional distance from other people are all effective preventative measures against contracting Covid19 (1 metre or 3 feet). Each country has had a consistent increase in confirmed cases. More stringent safeguards and strategies have already been put in place in several places in the event of the spread of such a virus.

Predicting how many instances of COVID-19 a hospital will see is crucial for making the most efficient use of its resources and developing effective treatment plans for patients who have contracted the virus. In wide range of applications, especially in business and academia, deep learning and machine learning are emerging as a promising study subject in recent days [7-8].

In [9], four supervised machine learning techniques are used to forecast the future of COVID-19: the Support Vector Machine (SVM), the Exponential Smoothing (ES) algorithm, the LASSO regression strategy, and the linear regression algorithm. It is shown that ES outperforms competing models in predicting the total number of fatalities, survivors, and contaminated cases. ES incorporates historical data prediction information for processing time series data. This establishes ES as a viable option.

A Modified Auto-Encoder (MAE), an AI-based method, is developed in [10] to predict new and cumulative mortality and verified COVID 19 cases under diverse intervention situations in real time in over a hundred different nations. This method falls short of expectations in both prediction accuracy and computational complexity.

The remainder of the paper is arranged as follows: The second section details the various prediction models used in the aforementioned studies. Section 3 outlines recommended technique. Section 4 presents experimental findings and performance comparison. Finally, Section 5, ends the study and outlines future research options.

2 Literature review

The WHO has approved COVID 19 outbreak data for confirmed patients in the next 10 days is used to evaluate this model. Root Mean Squared Relative Error (RMSRE), coefficient of determination (R2), Mean Absolute Percentage Error (MAPE), and computation time are all areas in which the FPASSA-ANFIS model [11] excels.

Long Short-Term Memory (LSTM) networks were created by Chimmula and Zhang (2020) to predict future pandemics like the current COVID-19 outbreak in Canada and worldwide. The COVID-19 data used in this study was collected by the Johns Hopkins and Canadian Health Boards and will be made available by March 31, 2020. Every day, statistics are kept on the number of deaths and the number of people who survive their injuries.

By using a wavelet modification, we are able to preserve time-frequency detail while simultaneously suppressing random fluctuations in the data. Then, COVID-19 time predictions are made using a network of Long Short-term Memory (LSTM). We test the efficacy of our models on the remaining 20% of the COVID-19 dataset after training them on the remaining 80%. The improved prediction accuracy [12] is a direct result of the invented device.

To rapidly and accurately anticipate COVID-19 time series, Castillo and Melin (2020) developed a hybrid technique that blends the fractal dimension with fuzzy logic. Ten different countries' publicly available data sets were used to construct the fuzzy inference system using time series in a given time period. The proposed solution's viability for the projected values of the 10 countries was then evaluated over a range of time horizons.

A 10-day and 30-day prediction window were used to evaluate the proposed method. The average accuracy of 98% is remarkable considering the intricacy of the COVID-19 problem.

With the use of AI-inspired algorithms, Hu et al. (2020) created real-time projections of Covid-19's size, duration, and end time throughout China. Here, a stacked auto-encoder was tweaked to simulate the spreading dynamics of an epidemic. This model was then used to make predictions in real time about where in China new instances of Covid-19 would be reported. The WHO gathered the information from January 11th, 2020, to February 27th, 2020.

Here, latent variables in auto-encoder and clustering approaches were used to arrange the provinces/cities in a way conducive to investigating the transmission structure. The average inaccuracy for a multi-step forecast was 2.27 percent for a 7-step forecast, 1.64 percent for a 6-step prediction, 2.08 percent for a 9-step forecast, 0.73 percent for a 10-step forecast, and 2.14 percent for an 8-step forecast. [14].

A deep learning-based prediction model for the COVID-19 outbreak in Saudi Arabia was proposed by Elsheikh et al (2020). To predict the total number of confirmed cases, total recovered cases, and total deaths in Saudi Arabia, this research used the Long Short-Term Memory (LSTM) network as a complete deep learning model. Official data was used to train this model. The model's parameters were fine-tuned by determining the optimal values for optimum prediction accuracy.

In order to determine the model's rightness, seven standard statistical factors are utilised [15]: the Root Mean Square Error (RMSE), the Coefficient of Variation (COV), the Coefficient of Residual Mass (CRM), the Overall Index (OI), the Mean Absolute Error (MAE), and the Effectiveness Coefficient (EC).

Around 6% minimization in RMSLE and improvement in absolute Pearson Correlation to 0.998 from 0.9978can be achieved as shown in results when compared with baseline models.

3 Proposed methodology

For forecasting COVID-19, we suggest using two methods such as Cat Swarm Optimization (CSO) for reducing the inertia weight linearly and then artificial intelligence i.e. CNN based partial binomial distribution is used. Figure 1 shows the proposed work's flow diagram.

3.1 Input

The input corresponds to COVID-19 forecasting dataset. From https://www.kaggle.com/davidbnn92/weather-data-for-covid19-data-analysis, this dataset is collected. Information like fog, temperature, fatalities, confirmed cases, date, long, lat, region, state, id are added in this dataset. From NOAA GSOD dataset, imported the weather data. Recent measurements are included using continuous update.

1. (a) **Data normalization using Min-max normalization**

In the normalising procedure, a mathematical formula is used to transform the original numerical values into a new set of range values. In the proposed study, we use min-max normalisation to the covid-19 dataset. Typically, min-max normalisation is used to level off data. All of the data in the dataset are normalised to fall within the given high and low bounds, and then the following expression is applied to the resulting data.

$$v' = \frac{v - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A \tag{1}$$

Where v is the previous value of each item in data, A is the attribute data, Min (A) is the smallest possible absolute value of A, and Max(A) is the largest possible absolute value of A. v' represents the updated value for each record in the data set, new max(A) the upper limit of the range, and new min(A) the lower bound (i.e essential boundary value range)

1. (a) **Feature selection using Linear Decreasing Inertia Weight based Cat Swarm Optimization (LDIWCSO) algorithm**

S.No	Authors name	Methods name	Merits	Demerits
	Al-Qaness et al (2020)	Modified ANFIS model	Higher classification accuracy and recall	It has Computational complexity
	Chimmula and Zhang (2020)	LSTM	No need of parameter tuning	LSTMs take longer to train LSTMs are easy to overfit
	Castillo and Melin (2020)	Fractal dimension and fuzzy logic scheme	It achieves 98% of forecasting average accuracy	Need to update the rules regularly
	Hu et al (2020)	Modified stacked auto-encoder	Higher True positive rate	It required joint training
	Elsheikh et al (2020)	Long Short-Term Memory (LSTM) network	Acceptable RMSE and MAE	It has sensitive to different random weight initializations

Table 1: Comparative analysis of existing Covid -19 forecasting approaches

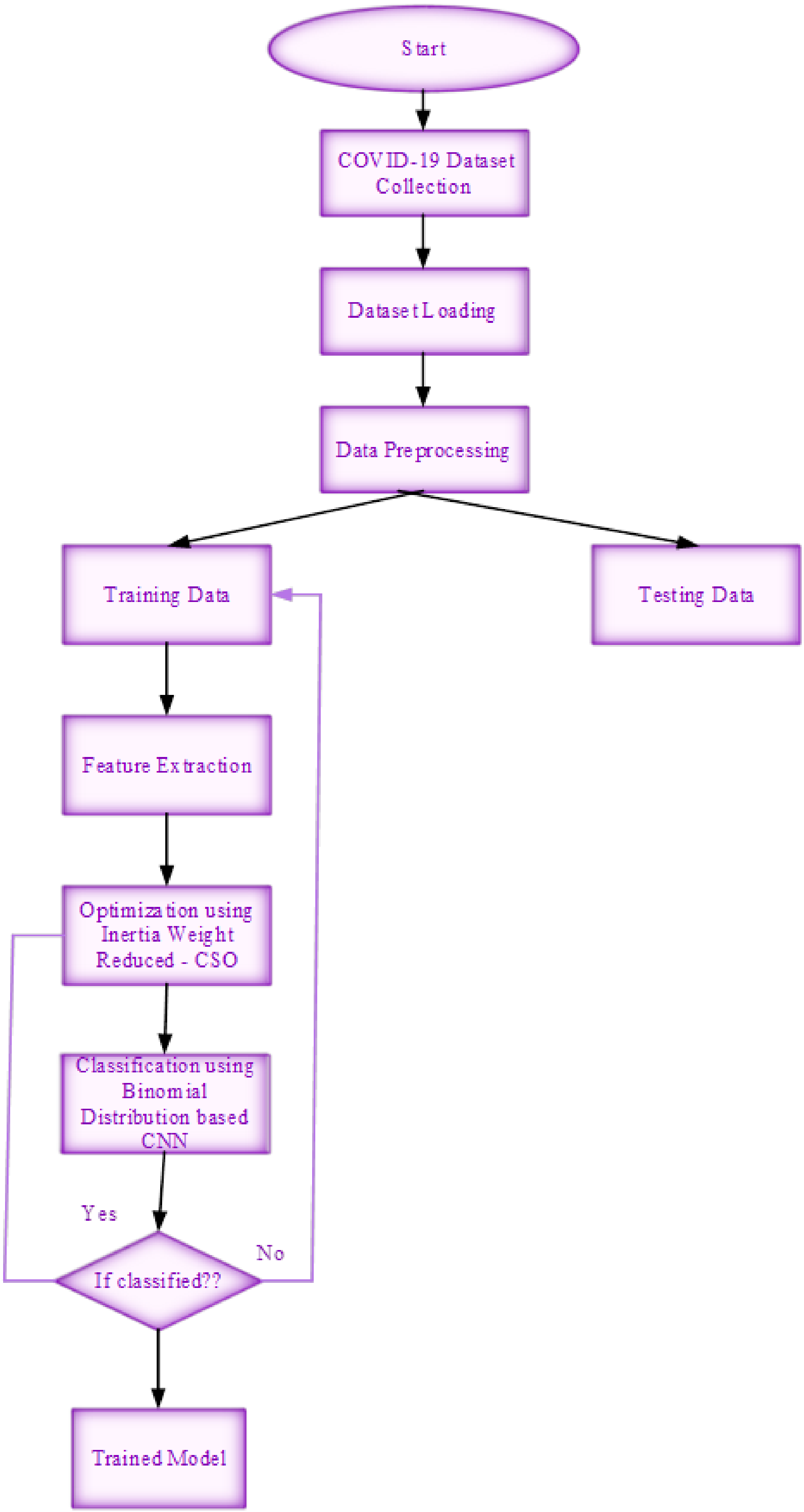

Figure 1: Flow diagram of the proposed work

For feature selection, we use the Linear Decreasing Inertia Weight based Cat Swarm Optimization (LDI-WCSO) method. The feline way of life inspired the creation of the Cat Swarm Optimization algorithm. This algorithm has both tracing and seeking search modes. Seeking mode simulates a cat's innate ability to maintain situational awareness even when at rest. In this mode, you may pretend to be a cat and follow and capture its prey. These frequencies are used to address optimization difficulties.
This suggested research effort takes features as input. Here, we characterise each feature in a population by giving it a fitness, speed, and location. A vantage point with rst to the collection of features that might be depicted by its location. For distances of dimension D, the corresponding scalar quantities are velocities. The fitness value is a measure of the quality of the solution set. In the suggested approach, accuracy of the classification serves as the fitness function.

$$Fitness function = Max(Accuracy) \tag{2}$$

CSO Process

CSO will assign different features to the tracing or searching mode in each generation. The placement of all elements was then rearranged. In the end, the feature with the highest accuracy value takes first place. The Cat Swarm Optimization (CSO) method is shown in Figure 2.
Mixture Ratio depicts the proportion of the cat's identifying characteristics (features) that are used for tracing (MR). The MR value will be rather low since cats spend much of their time testing. The steps involved.

1. Within a -dimensional space, N features are initialized with velocities and positions.
2. Features are distributed to tracing or seeking mode; features count in two modes defines MR.
3. Everyfeature's accuracy value is measured.
4. In every iteration, based on cat's mode, they are searched.Following subsection, describes the process involved in these two modes.
5. If terminal criteria are satisfied, algorithm is stopped; otherwise return to (**??**) for following iteration.

Seeking Mode

There are four different steps involved in the search mode: thinking about where you are in space, searching the region of the dimensions you've chosen, digging into your memory to find what you're looking for, and keeping track of how many dimensions have changed (SMP). SMP controls the amount of accessible searching memory. According to the SMP test, a score of 5 means that a cat can memorise and recall five distinct solutions.

A SPC is a boolean data type. If SPC is correct and also no changes are made to the solution set, then the information stored at a given address in memory will remain unchanged. The searching range's minimum and maximum values are determined by the SRD. In order to see how many various dimensions must be changed during the search strategy, the CDC is used. How to do a search is described in detail below.

Initial SMP copies are made depending on the latest feature position. If SPC is correct, one of these duplicates will maintain the current feature's location and be considered for selection right away. Additional adjustments must be made before they can even be considered contenders. Otherwise, all SMP copies will conduct searching, which will cause a shift in the system's overall position.

The % dimensions of CDC are randomised, causing a shift in the location of all copies requiring a modification. As a starting point, we choose the CDC's dimensions percentage. By randomly altering each chosen dimension, the current SRD percent value is lowered or raised. Copies becomes viable options if modifications have been performed. (3) Every candidate's fitness value is computed via the fitness function.

Every candidate's probability of being selected is computed. The selected probability ($\mathrm{P_i}$) is equal to one, if all candidates have same fitness value, else, using expression (3), $\mathrm{P_i}$ is computed. Range of lies between 0 and SMP; Candidates selection probability is given by $\mathrm{P_i}$; Overall fitness values maximum value is represented as $\mathrm{FS_{max}}$ and minimum value is represented as $\mathrm{FS_{min}}$; Candidates fitness value is given by $\mathrm{FS_i}$.

If objective is to find solution set havingmaximum fitnessvalue, then FS_b= FS_{min}; elseFS_b= FS_{max}.Probability computation using this function produces better candidate a higher chance of being selected,and vice versa, as follows:

$$P_i = \frac{|FS_i - FS_b|}{FS_{max} - FS_{min}} \tag{3}$$

(5) Based on selected probability (P_i), one candidate is selected randomly. Current feature is moved to this position after selecting the candidate.

Tracing Mode.

Cats target tracing is represented using tracing mode, here same strategy is followed.

(1) Expression (4) is used for updating velocities.

(**??**) Current feature's position is updated using expression(5) as mentioned below:

$$v_{k,d}^{t+1} = v_{k,d}^{t} + r_1 \times c_1 \times \left(x_{best,d}^{t} - x_{k,d}^{t}\right) \tag{4}$$

d=1,2,..,D,

$$x_{k,d}^{t+1} = x_{k,d}^{t} + v_{k,d}^{t} \tag{5}$$

Where,
$x_{k,d}^{t}$and $v_{k,d}^{t}$are current feature 's position and velocities at iteration
$x_{best,d}^{t}$gives best solution set from $feature_k$in population
$c_1$is constant
r_1 is a random number and its value lies between 0 and 1

Cat Swarm Optimization Algorithm

Best algorithm used for computing best global solution is conventional CSO. But, for producing acceptable solution, long time may be consumed by CSO in rare cases. So, algorithm's convergence and performance may get affected by this. A new parameter termed as inertia weight (w) is added in the expression used for updating positions to solve this issue. In algorithm's tracing mode, a new velocity update form is used.

A Linear Decreasing Inertia Weight factor is introduced in this work and it has a linear decrement to w$_{min}$ from w$_{max}$ with respect to iteration's increment and they indicates final and initial values. Reformed expressions are mentioned below.

$$v_{k,d}^{t+1} = w * v_{k,d}^{t} + r_1 \times c_1 \times \left(x_{best,d}^{t} - x_{k,d}^{t}\right) \tag{6}$$

w= w_{max}-$\frac{w_{max}-w_{min}}{iter_{max}} \times k$ (**??**)

$$v_{k,d}^{t+1} = (w_{max} - \frac{w_{max} - w_{min}}{iter_{max}} \times k) * v_{k,d}^{t} + r_1 \times c_1 \times \left(x_{best,d}^{t} - x_{k,d}^{t}\right) \tag{7}$$

Where,
w gives inertia weight
w_{max}represents maximum weight value
w_{min}representsminimum weight value
$iter_{max}$indicates maximum iteration
k is a Constant

Algorithm 1: Cat Swarm Optimization (CSO) algorithm

1. Start
2. In dataset, features count is initialized
3. Velocity and parameters are initialized
4. While (t <= it_{max})
5. For every cat
6. Fitness value is computed
7. Next, best cat is selected and saved into memory.
8. if feature (X_i) = *seeking mode*
9. Seeking process is initiated
10. Else
11. Tracing process is initiated
12. Termination condition is checked; if satisfied; program is terminated; otherwise, Step 4 to 11 are repeated.
13. Fitness function is computed and cat with best finest value is preserved.
14. Selected features
15. Stop

3.4 Classification using Partial Binomial Distribution using CNN

For classification, this research work uses Partial Binomial Distribution using CNN. The CNN is a highly powerful deep network. In different areas, great efficiency is shown by this network, especially in computer vision. There are three types of layers in CNN, namely, fully connected, sub-sampling and convolution layer. A CNN typical architecture is shown in Figure 3. Following section explains the every layer type.

Convolution layer

In this suggested work, the chosen characteristics are provided as an input. These input characteristics are convolved with a kernel (filter) in this convolution. This convolution output between the kernel and the inputs

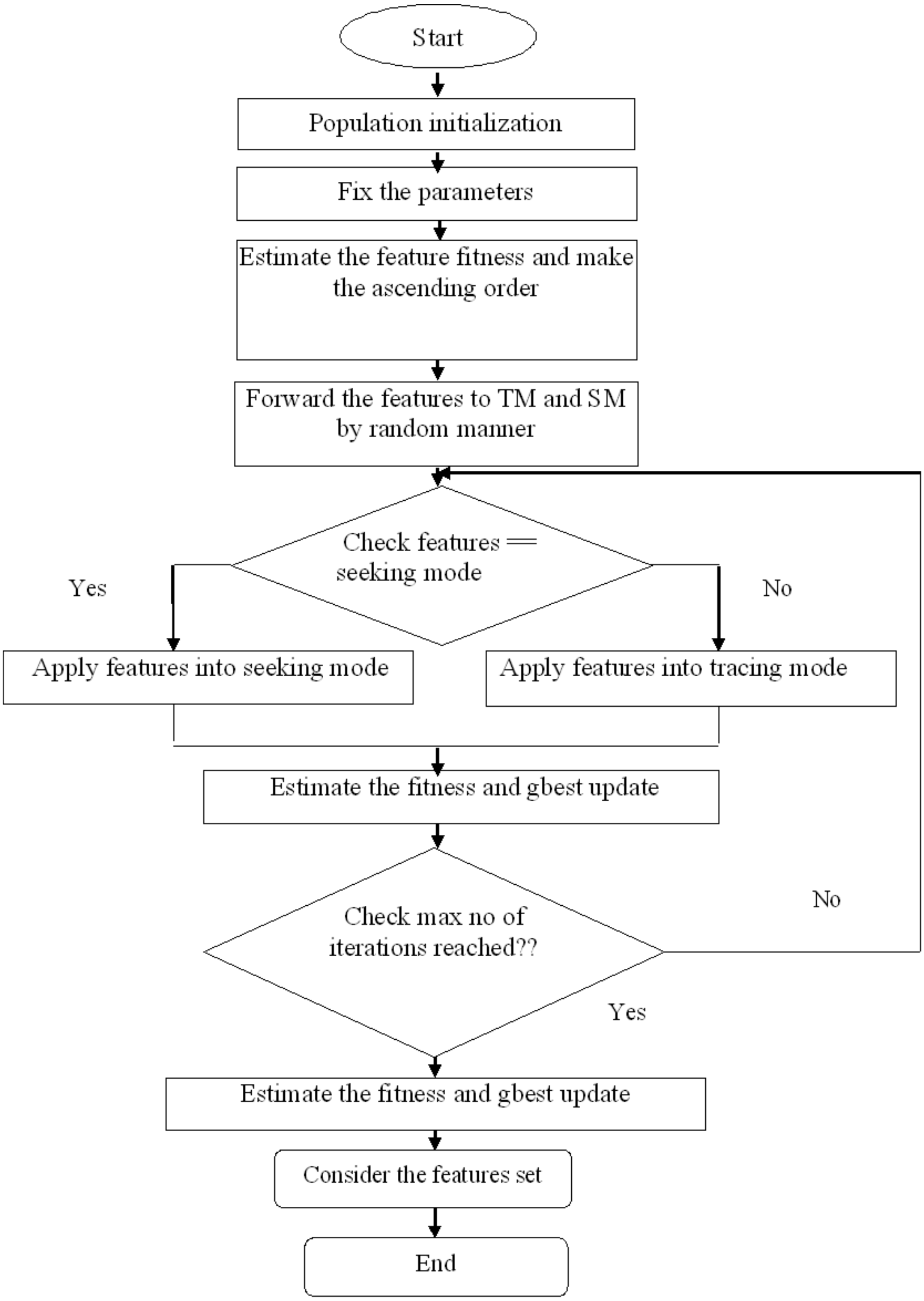

Figure 2: Workflow for CSO

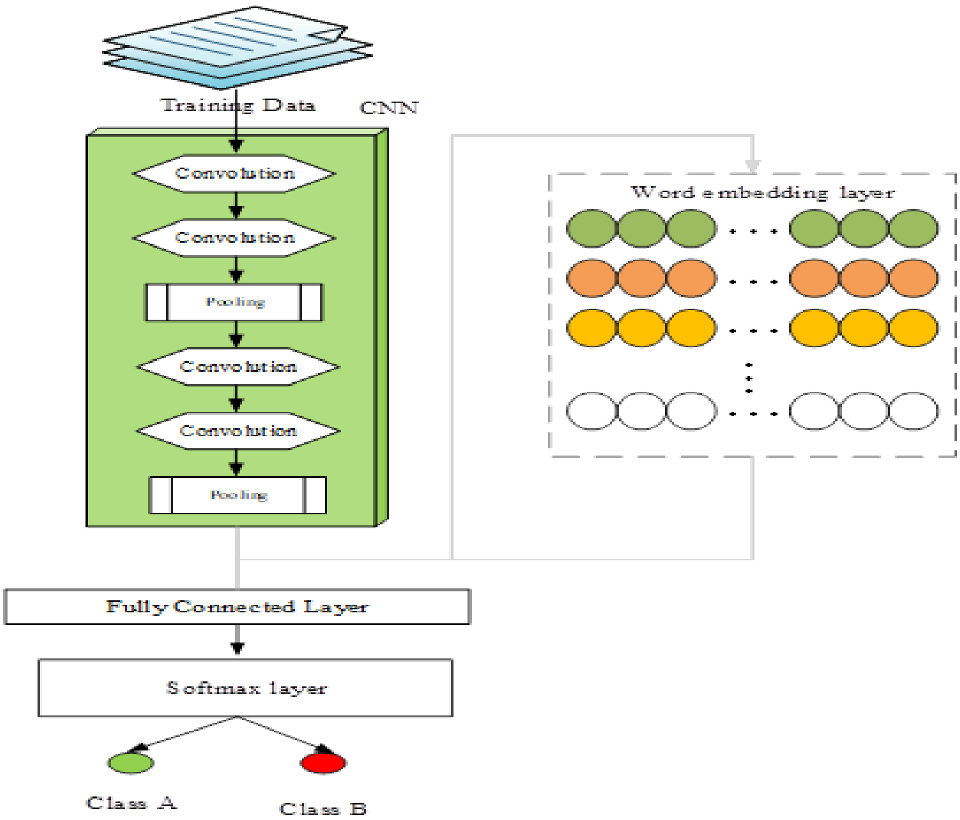

Figure 3: Convolutional Neural Network

features is then used to create the n output features. In this context, filter means the kernel of a convolution matrix, and a feature map of size i*i means the features computed by convolving the input with the kernel.

CNN consists of many layers of convolutional neural networks. Later convolutional layers take the feature vector as an input and output the same thing. A total of n filter bunches are used throughout all convolution layers. Several filtering are used to convolve the input, and the depth of the resulting feature map is proportional to the number of filters used in the convolution procedure ($n*$). Each filter map is treated as a unique feature at a predetermined input position.

$$C_i^{(l)} = B_i^{(l)} + \sum_{j=1}^{a_i^{(l-1)}} K_{i,j}^{(l-1)} * C_j^{(l)} \tag{8}$$

Where, bias matrix is given by $B_i^{(l)}$, kernel or convolution filter is represented as $K_{i,j}^{(l-1)}$ and its size is $a*a$. In same layer, i-th feature map is connected with feature map in layer $(l-1)$. Feature maps are available in output $C_i^{(l)}$ layer. In expression (9), input space corresponds to first convolutional layer $C_i^{(l-1)}$, that is$C_i^{(0)} = X_i$. Feature map is generated using the kernel. For convolutional layer output's nonlinear transformation, activation function is applied after convolution layer.

$$Y_i^{(l)} = Y(C_i^{(l)}) \tag{9}$$

Where,
$Y_i^{(l)}$represents activation function's output
$C_i^{(l)}$gives the input that it receives.
The rectified linear units (ReLUs), tanh, sigmoid are the commonly used activation functions. The ReLUs activation function is used in this work and it is represented as,

$$Y_i^{(l)} = \max(0, Y_i^{(l)}) \tag{10}$$

This function is extensively employed in deep learning models due to its effectiveness in reducing nonlinear characteristics and interaction. The output of ReLU is zero when the input is negative and the same input when the input is positive. One of the main benefits of this function is how quickly it can be trained compared to others. The error derivative of this function is negligible in the saturation area. The result is the elimination of future weight-status updates. The phrase "vanishing gradients issue" is used to describe this situation.

Pooling or Sub Sampling Layer

This layer applies a spatial dimension reduction to the feature map that was produced in the preceding convolution layer. Exactly this is one of the primary goals of this particular layer. This subsampling procedure is executed in the middle of the feature maps and the mask. There are a number of different subsampling methods that may be generated, including maximum pooling, sum pooling, and average pooling. Beyond convex hull of inputs $\{\mathrm{X}_{\mathrm{p}}^{\mathrm{c}}\}_{\mathrm{p}\in\mathrm{R}_{\mathrm{q}}}, Y_q^c$ is produced for CNN's better recognition.

Here, for simplicity, subscript q (output position) and superscript c (channel) are omitted. First, a binomial distribution is used for modelling local neuron activations $\{\mathrm{X}_{\mathrm{p}}\}_{\mathrm{p}\in\mathrm{R}}$ and this distribution has a standard deviation σ_x and mean μ_x.

$$\overline{X} \sim N\left(\mu_x, \sigma_x\right) \Longleftrightarrow \overline{X} = \mu_x + \epsilon\sigma_x \tag{11}$$

Where,

$$\mu_x = nX_S$$

$$\sigma_x^2 = \left(\ X_S - \mu_x\ \right)^2$$

$$\epsilon \sim N(0,1), \epsilon \in (-\infty,\ +\infty)$$

N indicates iterations count
X_S represents input features successfully beyond specific class

The expression (11) is modified according to the local pooling knowledge and it restrict the output to fall below mean μ_x as, Where, prior probabilistic model corresponds to half- binomial distribution $N_h(\sigma_0)$ and it has $\sigma_0 = 1$. For producing Y stochastically without the use of pooling parameter, formulated the fixed half-binomial pooling as shown in expression 13.

Fully Connected layer

The CNN's final layer is a traditional feed forward network and there may be one or more than one hidden layers. The Softmax activation function is used by output layer.

$$Y_i^{(l)} = f(z_i^{(l)}) \tag{12}$$

$$where z_i^{(l)} = \sum_{i=1}^{m_i^{(l-1)}} w_{i,j}^{(l)} y_i^{(l-1)} \tag{13}$$

Where, weights are represented as $w_{i,j}^{(l)}$ and each class's representation is tweaked using a single fully-connected layer, and the transfer function is modelled as f, while nonlinearity is modelled as a sigmoid. The neurons in a completely linked layer are preprogrammed for nonlinear behaviour. It is constructed independently in both pooling and convolution layers. The suggested CNN method is used to confirm deaths in certain time periods throughout India.

4 Experimental results

For proposed and available research techniques, in MATLAB, experimental evaluation is conducted. From https://www.kaggle.com/davidbnn92/weather-data-for-covid19-data-analysis, collected the COVID-19 forecasting data in this research work. The available Long Short-Term Memory (LSTM) networks, CNN, and proposed CNN method performance are compared in terms of f-measure, accuracy, recall, precision, and error rate values.

In Figure 4 we see the procedure of the suggested technique. In this proposed study, we take the COVID-19 forecast database as an input and normalise it using the min-max approach. You can see this procedure in Figure 5. Figure 6 depicts the suggested method for selecting the best attributes.

In the prediction of death and confirmed cases across India, training process of neural network is represented in Figure 7. Further, for detected result, decision may be incorrect (false) or correct (true), which is illustrated in figure 8. So, decision false under any one the following four possible classes, True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN).

The available Long Short-Term Memory (LSTM) networks, CNN, and proposed CNN method performance are compared in terms of f-measure, accuracy, recall, precision, and error rate values are represented in table 2.

1. **Accuracy**

Accuracy is a simple ratio of expected observations to total observations, making it the most straightforward performance metric.

$$Accuracy = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{FP}+\mathrm{FN}+\mathrm{TN}} \tag{14}$$

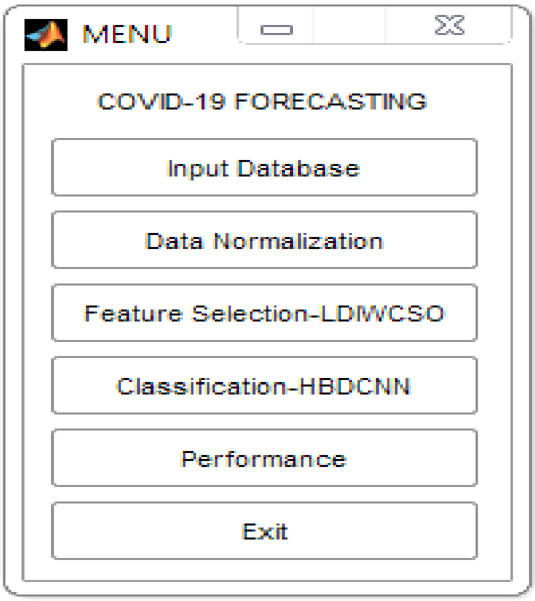

Figure 4: Layout for Proposed Method

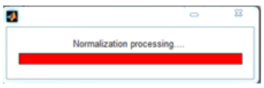

Figure 5: Normalization process

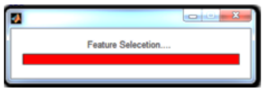

Figure 6: Feature selection

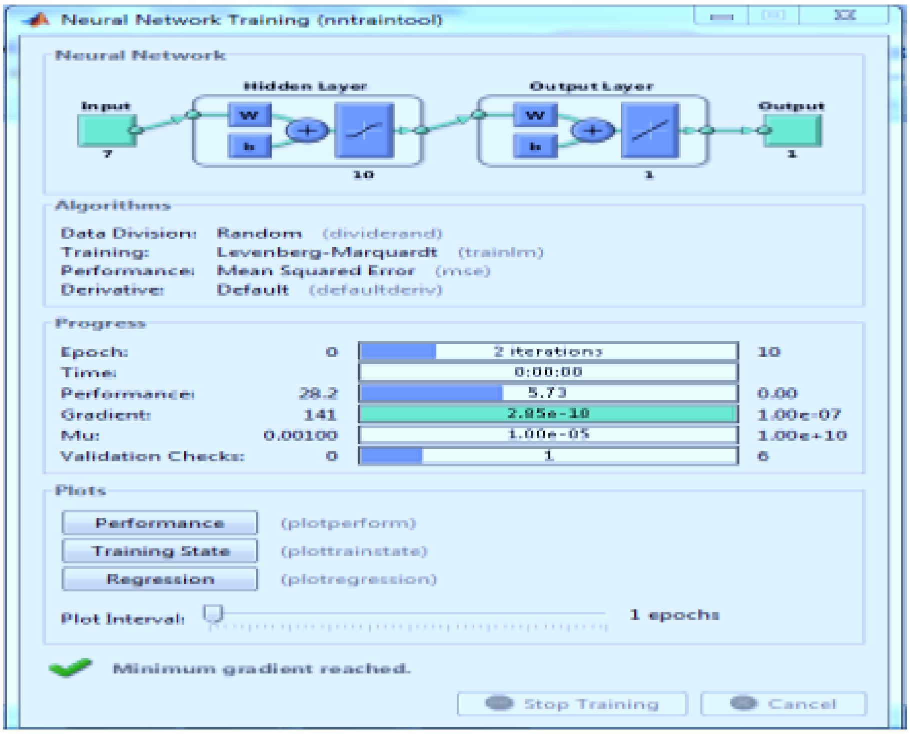

Figure 7: Neural network training

```
-------------------------------------------
             Actual Classes
-------------0---------1-----
Predicted|
  Classes|
_________|____________________
       0 |     9239.0 |     587.0
       1 |     761.0 |     7305.0
-------------------------------------------
-------------------------------------------
             Actual Classes
-----------------------------0--------1-----
True Positive              |   9239.00 |    7305.00
False Positive             |   587.00 |    761.00
False Negative             |   761.00 |    587.00
True Negative              |   7305.00 |    9239.00
Precision                  |   0.94 |    0.91
Recall or Sensitivity      |   0.92 |    0.93
Specificity                |   0.93 |    0.92
-------------------------------------------
Model Accuracy is 0.92
-------------------------------------------
```

Figure 8: Performance results

	LSTM	CNN	HBDCNN
Accuracy	87.9220	90.7445	92.5777
Precision	87.7028	90.5500	92.4192
Recall	87.8875	90.7531	92.5654
F- Measure	87.7950	90.6514	92.4922
Error Rate	12.0780	9.2555	7.4223

Table 2: Performance analysis

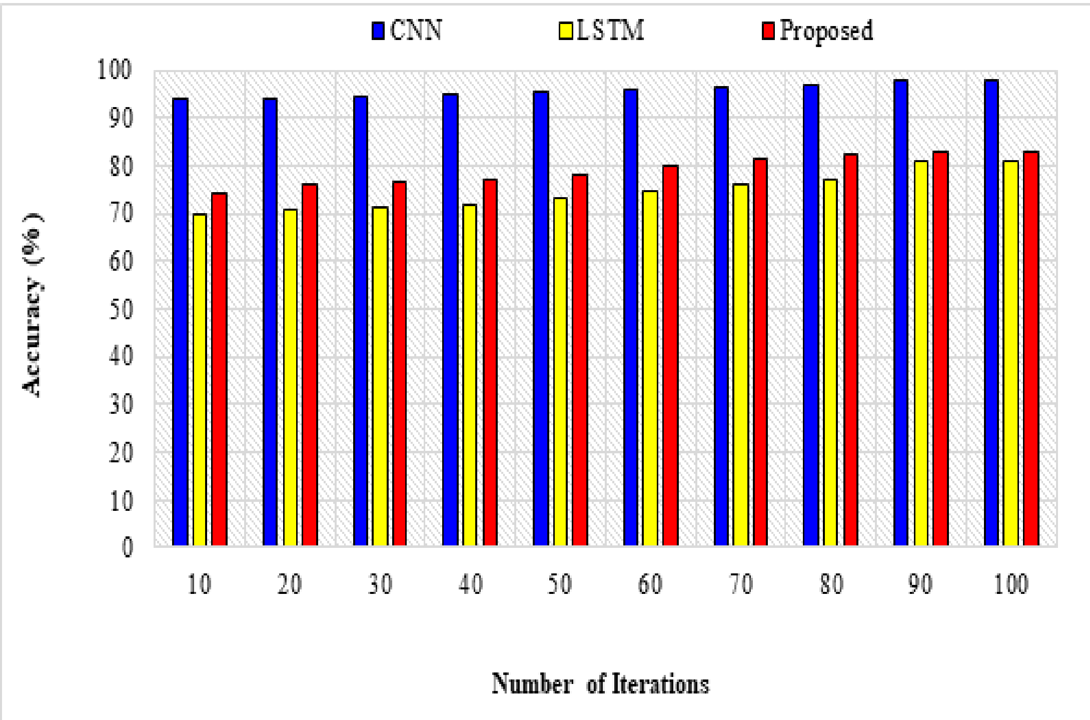

Figure 9: Accuracy comparison

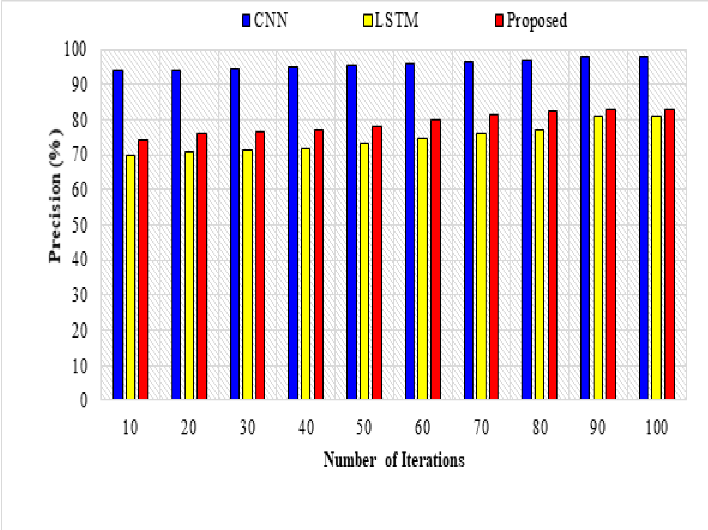

Figure 10: Precision comparison

The Proposed technique and available CNN and LSTM techniques are compared with each other in terms of accuracy and its results are shown in Figure 9. The proposed algorithm is used in this work for selecting optimum features. For covid-19 prediction, proposed method is used in this proposed work and this prediction is done using selected features. This produces better accuracy. Around 92.57% accuracy value is produced by proposed method, 90.74% is produced by CNN and 87.92% is produced by LSTM as indicated in experimental results. When compared with available techniques, high accuracy value is produced by proposed method as shown in experimental results.

1. **Precision**

The value of accuracy is defined as the proportion of accurately predicted positive observation relative to the overall proportion of expected positive observations.

$Precision = \frac{TP}{TP+FP}$	17

The proposed technique and available CNN and LSTM techniques are compared with each other in terms of precision and its results are shown in Figure 10. There are different techniques are represented in x-axis and in y-axis, precision values are represented. Around 92.41% precision value is produced by proposed method, 90.55% is produced by CNN and 87.70% is formed by LSTM as indicated in experimental results. When compared with available techniques, high precision value is produced by proposed method as shown in experimental results.

1. **Recall**

The recall ratio in covid-19 forecasting is the proportion of valid instances that were successfully recovered.

$$Recall = \frac{TP}{TP + FN} \tag{15}$$

The proposed technique and available CNN and LSTM techniques are compared with each other in terms of recall and its results are shown in Figure 11. Various techniques are represented in x-axis and in y-axis, recall values are represented. For covid-19 prediction, the proposed technique is used in this proposed work. Around 92.56% recall value is produced by proposed technique, 90.75% is produced by CNN and 87.88% is produced by LSTM as indicated in experimental results. When compared with available techniques high recall value is produced by proposed method as shown in experimental results.

1. **F1 score**

The F1 score is calculated by averaging the weighted recall and accuracy scores. As such, it may be used as a statistical metric for evaluating classifier performance. This score takes into account the number of false positives and negatives. Its F1 score is written as,

F-measure comparisons between the proposed method and existing CNN and LSTM methods are shown in Figure 12. Different methods are shown along the x-axis, while F-measure values are shown along the y-axis.

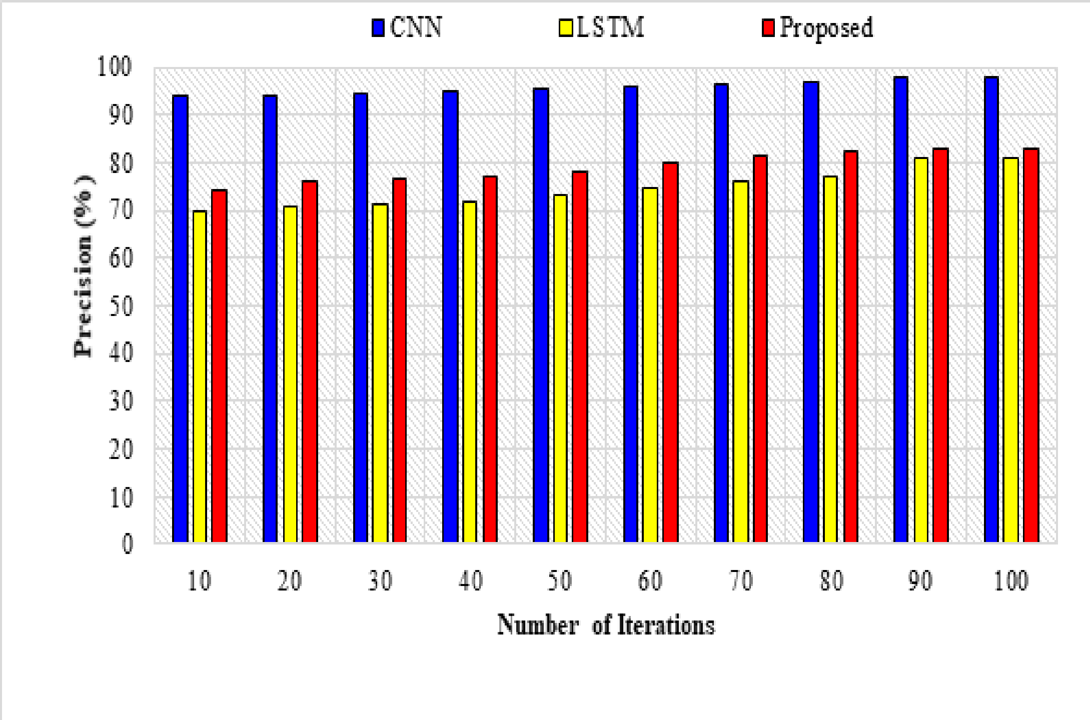

Figure 11: Recall comparison

$F-measure = 2\frac{Precision*Recall}{Precision+Recall}$	19

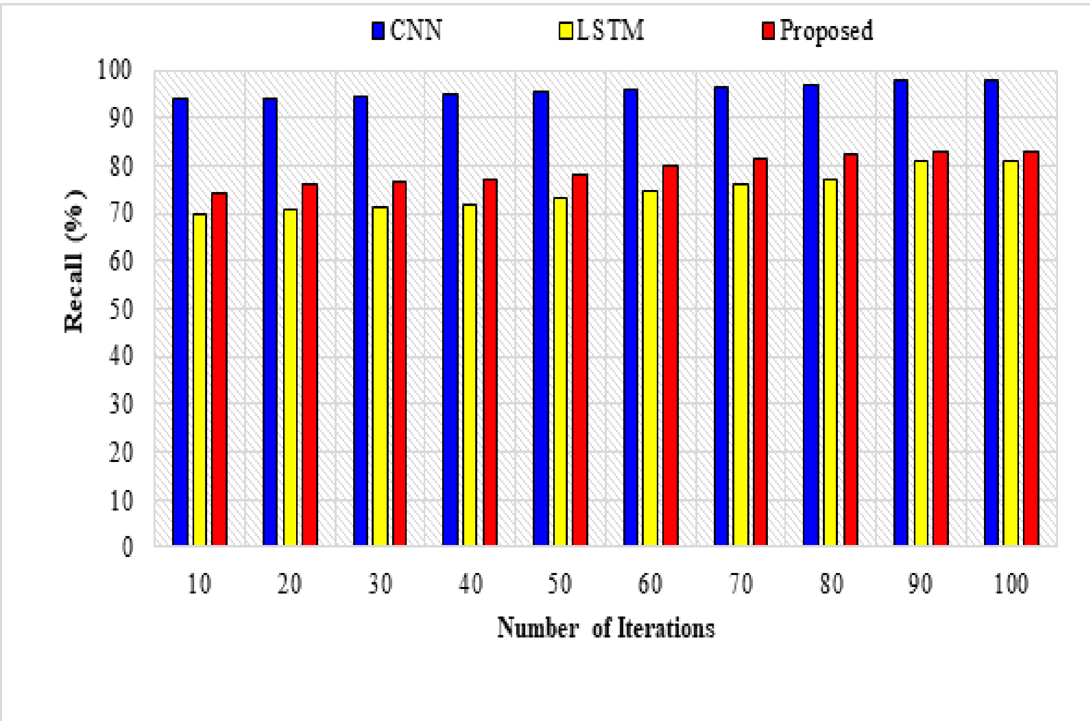

Figure 12: F-measure comparison

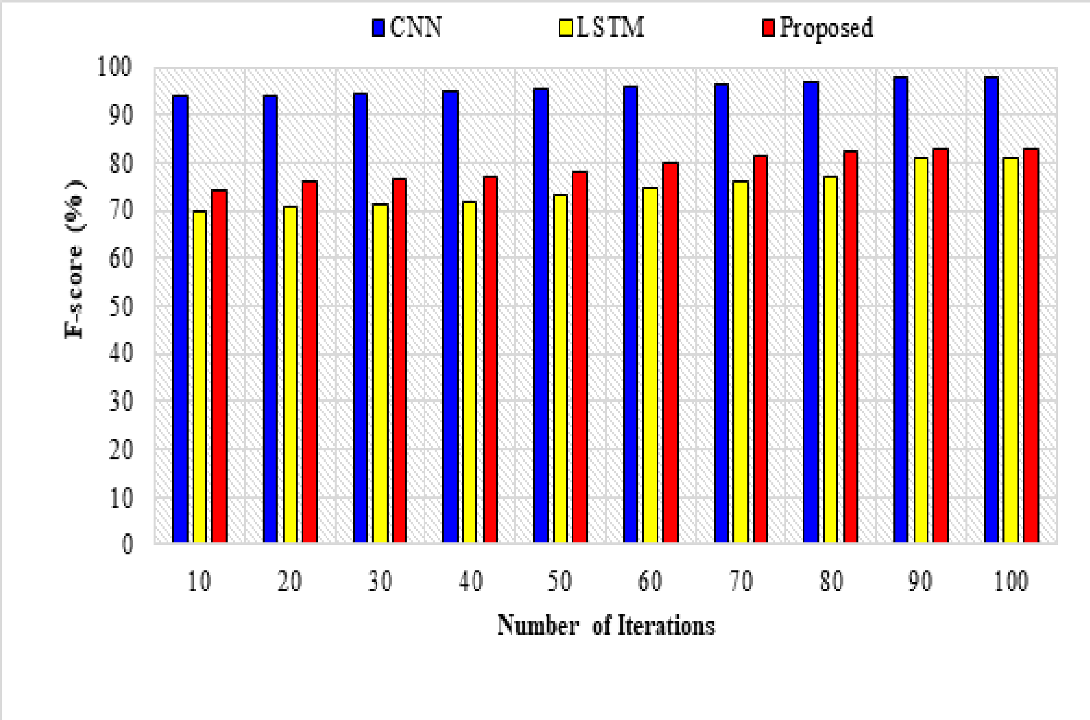

Figure 13: Error rate comparison

The experimental findings show that the suggested method yields an F-measure of around 92.49%, whereas CNN and LSTM yield only about 90.65% and 87.79%, respectively. Experimental findings reveal that the suggested method yields a high f-measure value, in comparison to existing methods.

1. **Error rate**

Figure 13 depicts the error rate performance of LSTM, CNN, and the suggested methods. The x-axis depicts the various methods, while the y-axis shows the associated error rates. According to the findings of the experiments, the proposed method results in an error rate of around 7.4%, whereas CNN results in a rate of about 9.25% and LSTM results in a rate of about 12.07%.

5 Conclusion

In this paper, we propose using AI with an optimization approach to forecast future deaths and confirmed cases. Min-max normalisation is used to first adjust the data. The optimal features for a given classification task are chosen via an optimization method. Predictions of covid-19 based on these characteristics are made using a CNN trained with a binomial distribution. According to the testing findings, the suggested system achieves an accuracy of approximately 92.57%, precision of about 92.41%, recall of about 92.56%, f-measure of about 92.49%, and error rate of about 7.4%. Improvements in classification accuracy may be possible in the future with the use of several optimization methods like Ant Colony Optimization (ACO), Simulated Annealing (SA), and Harmony Search (HS).

References

[1] Elsheikh, A.H., Saba, A.I., Elaziz, M.E., Lu, S., Shanmugan, S., Muthuramalingam, T., Kumar, R., Mosleh, A.O., Essa, F.A., & Shehabeldeen, T.A. (2020). Deep learning-based forecasting model for COVID-19 outbreak in Saudi Arabia. Process Safety and Environmental Protection, 149, 223 - 233.

[2] Chintalapudi, N., Battineni, G., & Amenta, F. (2020). COVID-19 virus outbreak forecasting of registered and recovered cases after sixty day lockdown in Italy: A data driven model approach. Journal of Microbiology, Immunology, and Infection, 53, 396

[3] Lukman, A.F., Rauf, R.I., Abiodun, O.E., Oludoun, O., Ayinde, K., & Ogundokun, R.O. (2020). COVID-19 prevalence estimation: Four most affected African countries. Infectious Disease Modelling, 5, 827 - 838.

[4] Yan, C.H., Faraji, F., Prajapati, D.P., Boone, C.E., & Deconde, A.S. (2020). Association of chemosensory dysfunction and COVID-19 in patients presenting with influenza-like symptoms. International Forum of Allergy & Rhinology, 10, 806 - 813.

[5] Almaghaslah, D., Kandasamy, G., Almanasef, M., Vasudevan, R., & Chandramohan, S. (2020). Review on the coronavirus disease (COVID-19) pandemic: Its outbreak and current status. International Journal of Clinical Practice, 74.

[6] Khajanchi, S., & Sarkar, K. (2020). Forecasting the daily and cumulative number of cases for the COVID-19 pandemic in India. Chaos, 30.

[7] He, S., Tang, S., & Rong, L. (2020). A discrete stochastic model of the COVID-19 outbreak: Forecast and control. Mathematical biosciences and engineering : MBE, 17 4, 2792-2804 .

[8] Kirbas, I., Sözen, A., Tuncer, A.D., & Kazanc–oğlu, F.Ş. (2020). Comparative analysis and forecasting of COVID-19 cases in various European countries with ARIMA, NARNN and LSTM approaches. Chaos, Solitons, and Fractals, 138, 110015 - 110015.

[9] D. N. V. S. L. S. Indira, Rajendra Kumar Ganiya, P. Ashok Babu, A. Jasmine Xavier, L. Kavisankar, S. Hemalatha, V. Senthilkumar, T. Kavitha, A. Rajaram, Karthik Annam, Alazar Yeshitla, "Improved Artificial Neural Network with State Order Dataset Estimation for Brain Cancer Cell Diagnosis", BioMed Research International, vol. 2022, Article ID 7799812, 10 pages, 2022. https://doi.org/10.1155/2022/7799812.

[10] Hu, Z., Ge, Q., Li, S., Jin, L., & Xiong, M. (2020). Evaluating the effect of public health intervention on the global-wide spread trajectory of Covid-19. medRxiv.

[11] Al-qaness, M.A., Ewees, A.A., Fan, H., & Abd El Aziz, M.A. (2020). Optimization Method for Forecasting Confirmed Cases of COVID-19 in China. Journal of Clinical Medicine, 9.

[12] Karthik, A., MazherIqbal, J.L. Efficient Speech Enhancement Using Recurrent Convolution Encoder and Decoder. Wireless Pers Commun 119, 1959–1973 (2021). https://doi.org/10.1007/s11277-021-08313-6.

[13] Castillo, O., & Melin, P. (2020). Forecasting of COVID-19 time series for countries in the world based on a hybrid approach combining the fractal dimension and fuzzy logic. Chaos, Solitons, and Fractals, 140, 110242 - 110242.

[14] Hu, Z., Ge, Q., Li, S., Jin, L., & Xiong, M. (2020). Artificial Intelligence Forecasting of Covid-19 in China. International Journal of Educational Excellence.

[15] Elsheikh, A.H., Saba, A.I., Elaziz, M.E., Lu, S., Shanmugan, S., Muthuramalingam, T., Kumar, R., Mosleh, A.O., Essa, F.A., & Shehabeldeen, T.A. (2020). Deep learning-based forecasting model for COVID-19 outbreak in Saudi Arabia. Process Safety and Environmental Protection, 149, 223 - 233.

Cite this paper as:

Murugesan, R. K.; Madhu, K.; Sambandam, J.; Malliga, L. (2023). Covid-19 Forecasting Using CNN Approach With A Halbinomial Distribution And A Linear Decreasing Inertia Weight-Based Cat Swarm Optimization, *International Journal of Computers Communications & Control*, 18(1), 4396, 2023.
https://doi.org/10.15837/ijccc.2023.1.4396