International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

IJISAE

ISSN:2147-6799 www.ijisae.org Original Research Paper

A Stacking Model for Outlier Prediction using Learning Approaches

Mr. Boddu L. V. Siva Rama Krishna*[1], Dr. V. Mahalakshmi[2], Dr. Gopala Krishna Murthy Nukala[3]

Submitted: 17/07/2023 **Revised**: 09/09/2023 **Accepted**: 22/09/2023

Abstract: Outliers are considered as unexpected things observed while analyzing the data. Investigators found that the prediction and identification of outliers are extremely complex. Generally, a stream is measured as an unbounded data source executed promptly, and this research provides a novel way of predicting the outliers over the incoming data. Here, the incoming data is acquired from the hospital to validate the patients' records. There are higher chances of outliers over the incoming data based on the density of the arrival data. The execution of outlier prediction is performed independently with the integration of LSTM (Long Short Term Memory) over the stacked CNN model. The outlier detection process constantly measures the incoming data from the emotional input as an outlier or inlier. Here, the data reconstruction is achieved with the auto-Regressive model, and the prediction model considers the outliers to construct the training data. Various hidden representation acquired from the stacked model is considered for outlier prediction, and the experimental results demonstrate that the anticipated model shows superior prediction accuracy, specificity, sensitivity and F1 score. The proposed model is best-suited for prediction as the deep learning approaches perform well over complex applications and acquire superior results. The simulation is done in MATLAB 2016b, and the performance metrics show a better trade-off than the other approaches.

Keywords: *Outliers, Deep Learning, Auto-Encoder, CNN, Stacking*

1. Introduction

One of the important tasks in the statistics and data mining fields is the detection of outliers. There is an observation which is an outlier with the value to deviate from the general and consider this doubtful. The systematic errors are found and revealed with the data in the data analysis process, which is the primary task in analyzing outliers. The analysis of outliers is important to gain in the broad applications in the domains such as credit card, tax, cyber security, insurance, healthcare, military, and so on. The techniques of the outlier detection are shown, and the

definition for the normal behaviour or the region to each possible consideration in all the factors is also a trick. Varun Aggarwal et al. [1] presented that the normal boundary for the outlying data to define is very enigmatic due to the slight alter between the two points. The models to detect the outlier have varied designs for various applications concerning different requirements and constraints that are very particular to the domain. The factors are provided concerning the detection of outliers presented by Chen et al. [2] and the data nature, domain of

application; also, it is the discipline of knowledge. The technique to detect the outlier is to identify the abnormal patterns from the data, which is input. The input data nature instances consist of multiple features of various kinds such as binary, categorical, and univariate or multivariate, that is, single data types or multiple data types. The approach to detecting the outlier is essential in choosing the better features for the algorithm from the input data to provide better outcomes. Many predictive models use the labelled data for training apart from the nature of data in which the labels have normally defined the outlier or normal.

De Stefani et al. [3] and Ferdowsi et al. [4] stated that there are three approaches divided into unsupervised, Semi-Supervised, and supervised detection of outlier approaches that are turned on the labels to an extent. The approach to detect the supervised outlier: File et al. [5] stated that the technique to implement the correct predictive model is the instances of fully labelled data classified as normal or an outlier. It needs more effort to obtain the labelled data, which is costly like it is gathered from human experts. The approaches, methodologies, model or system differs depending on the applications of the real world [6] – [8]. In the same way, these algorithms to detect the outlier are compared to lie on a few particular metrics that test the qualities of the approach. Kiani et al. [9] discussed the short review across the algorithm to detect the outlier with the empirical analysis. It is concluded that the classification and the clustering depend on the algorithms to detect the outlier, which is highly scalable and effective with low computation cost. On the other hand, the depth and statistical-oriented sliding window are at the least stage. The effective, less cost computation is the distance-based outlier extended to the higher dimension data. Thus, there is a need for a novel method to predict the outlier. Here, a stacked network model is proposed, the prediction based on medical data is implemented, and the model gives better outcomes than other approaches.

[1] Research Scholar , Department of Computer Science and Engineering, Annamalai University,Chidambaram-608002,India
[2] Assistant Professor, Department of Computer Science and Engineering, Annamalai University,Chidambaram-608002,India
[3] Professor, Department of Information Technology, SRKR Engineering College,Bhimavaram-534204,India
** Corresponding Author Email: krishna2928@gmail.com*

The work is structured as follows: section 2 gives a wider analysis of various existing approaches; section 3 explains the methodology. The numerical outcomes are provided in section 4, and the conclusion is provided in section 5.

2. Related Works

The clinical alerting system is designed to detect adverse events as soon as possible. The technique of general alert utilizes the rules to identify the data of EMR for the patient to the particular condition of the clinic or the group of expert redefined criteria physiologically. The alert will raise the clinic's patient [17] if the data find patterns. The signal alert takes different forms like the popup windows on the interface of EMR or the messages such as paging or email to send to the patient's physician. The altering systems are explored in an extensive both commercially [21] to [23] and academically [18] to [20]. Various tasks are used, such as detection of deviation from the infected protocols for treatment of disease [22], growth disorders detection [19], adverse drug events detection [24] and [25] and detection of essential events in people with diabetes clinically [18] and management of the congestive failure of heart [26].

Most general kinds of altering systems and monitoring systems utilize the rules, which are constructed manually using the clinical domain experts with the help of personal experience and knowledge. The rules and the benefits generate the alerts, which are obtained from the experience of the experts directly. Consider an instance has rules to screen for the allergies of drug and the interactions and also the rules to diagnose the syndromic such as doubted sepsis, and acute lung injury and the estimation of the severity of disease [27]. Moreover, the altering systems based on knowledge depend on rules such as to use at current that needs manual construction that is difficult and time-consuming. Additionally, the experts define the rules using the design and the restricted coverage of the huge adverse events space, specifically more complicated events in an adverse way. In addition, the rules are based on knowledge for detecting and monitoring, which is implemented explicitly. Lastly, there is difficult to tune the rules to obtain better performances clinically in the circumstances where it is deployed. The rules alert are not common to implement with the expert's considerable effort to retire such as turned off which are short because of the large fake rates of alert after the deployment [28], and [29], due to the fake fatigue alert, avoids these alerts when these rules are still active [28] to [30]. Then, the rules are designed by the performance in a careful way that requires to adapt and optimization for obtaining positive clinical outcomes having the reduction of fake alarms that are the important aim.

A new technique to detect medical error is developed and pioneered in the latest research [8], [9], which complements the alert technique based on knowledge. A raise of alert for the decisions of clinical care in a brief manner is great, not usual, as unconventional in terms of previous patients having similar or identical conditions. This technique for the rationale is that the previous records of the patients have the majority in the EMR, which reflects the standard local care of clinics. This care deviated from these standards, like the medication decision to detect and correspond to the frequent errors for justifying the alert. An important merit of the alerting technique based on anomaly is the methods based on data-driven knowledge that does not depend on the altering rules and monitoring that experts implement. Apart from this, the technique based on outliers is driven using the deviation from regular clinical care patterns. The technique is considered by the features applicable to the clinic's broad range of conditions and circumstances, and coverage of the clinical alert is deep and broad, not for the knowledge-based systems. The alerting and monitoring based on outlier has the effort for complementing the usage of alerting systems based on knowledge deployed presently to improve the complete coverage of clinic of the present systems of alert.

3. Methodology

This section provides a detailed explanation of the proposed stacking model. They are pre-processing, feature reduction, and classification with the deep network model. Finally, metrics like accuracy, precision, F1-measure, recall, etc., are evaluated. Fig 1 depicts the block diagram of the anticipated outlier prediction model.

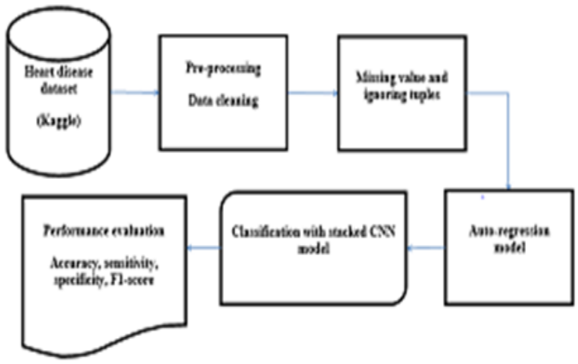

Fig 1 Workflow of the outlier prediction model

3.1 Dataset acquisition

This work considers the Kaggle dataset for heart disease prediction, which includes data from four databases: Switzerland, Hungary, Cleveland, and long beach. The dataset comprises 76 prediction attributes; however, all published experiments include 14 subsets. The target field specifies the occurrence of heart disease for the patients with the integer values, i.e. 0 for no disease and 1 for disease.

(https://www.kaggle.com/datasets/johnsmith88/heart-disease-dataset). The attribute information includes age, sex, blood pressure, chest pain type, cholesterol level, fasting sugar, resting ECG results, maximal heart rate, angina due to exercise, old peak, the slope of peak exercise, vessels coloured, thal value (normal or fixed).

3.2. Pre-processing

Here, data cleaning is performed to eliminate the missing and irrelevant parts. It includes both noisy data and missing data. Noise is considered meaningless data that the machine cannot analyze. It occurred due to the error data entries and fault data collection, which can be smoothened using the binning method. The data is partitioned into segments of equal size, and the process is completed. Every segment is handled separately, and the data is replaced using its boundary and mean values to complete the task. The missing data is identified when some data is missed in the provided dataset. It is handled in diverse ways: 1) ignoring the tuples and filling the missing values. Formerly, the dataset was quite larger, and multiple values were missing from the data; the latter is filled with the missing values manually using the probable value and attribute mean.

3.3. Auto-regressive model

Here, auto-regression is adopted to predict the cases of heart disease and the related characteristics are identified. Initially, the data over the affected patients are converted to time series data, and the correlation among the features is identified based on the edge between the features. It is provided to predict the correlation among the features. Considering N as the number of patients, the log difference of total affected patients, sit, is represented as in Eq. (1):

$$s_{it} = \delta \log\left(\frac{affected\ (i,t)}{affected\ (i,t-1)}\right) \tag{1}$$

Here, δ is constant (hyper-parameter), and the optimal value is set as 0.5.

$$s_{it} = \frac{1}{2}\log\left(\frac{affected\ (i,t)}{affected\ (i,t-1)}\right) \tag{2}$$

$$s_{it} = \log\left(\sqrt{affected\ (i,t)} - \log\left(\sqrt{affected\ (i,t-1)}\right)\right) \tag{3}$$

Here, affected (i,t)specifies total affected cases from i data at t time. The auto-regressive model is provided as:

$$s_{it} = \propto_0 + \propto_1 n_i^{-1} \sum_{j=1}^{N} a_{ij} s_j(t-1) + \propto_2 s_{i(t-1)} + \epsilon_{it} \tag{4}$$

Here, 'N' represents total patients, s_it represents log difference of affected cases from i data at t time; A=(a_ij)_(N*N) represents correlation matrix among the log-returns; n_i represents adjacent matrix A and ϵ_it represents a normal distribution.

3.4. Stacking CNN Model

A theoretical classification network is initiated in the proposed system based on the modified CNN adjoined to the secondary LSTM phase for classifying the provided input, which is one of the four severe scores from 0 to 3. Fig 2 shows the simple pipeline for the suggested system. The auto-encoder block passes the severe feature to the various branches of convolution to be considered in the top portion, which exhibit the suggested phase of CNN that originated from the provided data, and the separate convolutional branch is blocked for the extraction of edge dominant features correspondently. Then, the block of LSTM in a sequential way for every data represented in the bottom area for performing the last predictions depends on the created joint weight vectors from the suggested blocks of LSTM and CNN.

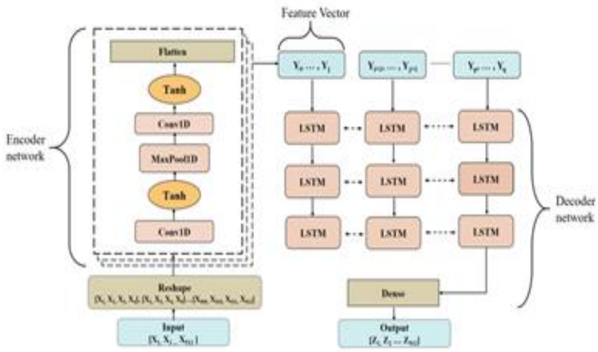

Fig 2 Deep network architecture

The images are passed primarily to the block of auto-encoder in the proposed system for lowering the noise and then pull out the powerful features that are found as the better discrimination characteristics among the scores. The auto-encoder is introduced to consider the input $x \in [0,1]^d$ to map to the representation of latent $y \in [0,1]d'$ having the occurrence of the mapping via the function $y_i = s(Wx_i + b)$. The presentation is hidden to mapped again to the similar shape reconstruction through $z_i = s(W'Y_i + b')$ as input. The non-linear function is s, such as the sigmoid function. The encoder is the first part, and the decoder is the second. This model has the parameters to optimize in the manner that the average error of reconstruction is low. The proposed system has the input images that are defined as having the de-noising auto-encoder assistance that gives the hidden layers for discovering the considered features from any image as input. The input size is encoded to feature maps, representing the input's latent space. The conventional blocks of the auto-encoder are used for one of the important issues is the nature of the prominent features to erase. The interconnection between the decoded and encoded features to resolve this issue to add via the combination for preserving the features to rule and the resemblance is kept between the output and input similarly is utilized in the segmentation of supervised networks like FCN (Fully

Connected Network). In the proposed system, the decoding part is established below the presentation.

$$z_i = s(W' z_{i-1} + b') \oplus y_i, \;\; i = 1,2,3 \;\; where\ z_0 = y_0 \tag{5}$$

Where a symbol indicates the concatenation ⊕, Fig 2 shows the branch of auto-encoder for the best understanding of the establishment information, as every step is presented. The auto-encoder is entered for the branch, for the input size is twisted to the size matrix that results in the size given to the branch.

Fig 2 shows the branch of auto-encoder that the primary three operations of encoding create the size for the encoded feature matrix y_0 that is placed in the mid of the figure, and thus consecutive three operations of decoding are carried out. s(W^' z_0 +b') is generated using the z_0 = y_0 to obtain the primary output, decoded output and concatenate y_1. It is important to consider that the part of encoding, operations of max pooling, and convolution are performed for the part of decoding, operations of upsampling, and convolution are carried out that is presented in Fig 2 that is relevant arrowheads to denote. The transformation of y_0 to the matrix size follows the operations of upsampling and convolution to concatenate with the previous branch of encoded y_1, which is also a similar size. The convolution, concatenation, and the upsampling process are repeated till the size of the feature matrix is obtained as 128× 128× 16. The size of the filter is altered to 16 rather than 8 at the earlier encoding stage, which creates the size of the matrix 128 × 128 × 16. The output size is reconstructed to obtain 128× 128× 3 to follow the deconvolution. The error of reconstruction is minimized by using the loss function. The proposed system incorporates the categorical cross-entropy loss function [22] to use the softmax or sigmoid activation function. The definition of the cross-entropy function is provided below.

$$CE = -\sum_{i}^{N} t_i \log f(s)_i \tag{6}$$

Here, the total number of classes is N, the relevant field is t, and the softmax function is f(s), which is presented below.

$$f(s)_i = \frac{\exp(s_i)}{\sum_j^N \exp(s_j)} \tag{7}$$

For auto-encoder, entropy function is diminished to:

$$L_H\,(x,z) = -\sum_{k=1}^{N} [x_k \log(z_k) + (1 - x_k) \log(1 - z_k)] \tag{8}$$

3.5. A special convolution branch is initiated for classification rather than the conventional autoencoder. The depthwise separable convolution is used in [23] instead of the traditional CNN in the block of based CNN. Three separate kernels are utilized in the depthwise separable convolution rather than the single size kernel 3 × 3 × 3. Every size of the kernel has 3 × 3 × 1. Every kernel twist has 1 input layer channel. Every convolution gives the size map of (M - 2) × (N - 2) × 1 for the size of input M × N × 3. The size (M - 2) × (N - 2) × 3 is achieved to stack these maps altogether. The convolution 1 × 1 is used with the size of kernel 1 × 1 × 3 for extending the depth in the second stage of the depthwise separable convolution. The input image having the size to convolve (M - 2) × (N - 2) × 3 with every kernel 1 × 1 × 3 gives the size map as (M - 2) × (N - 2) × 1. Then, the layer's size to obtain as (M - 2) × (N - 2) × K after using the K number of convolutions 1 × 1. The complete operation of the convolution obtains less time over the time considered using the conventional convolution as the separable convolution is divided by the operation of convolution to the pointwise convolution and the depth-wise convolution. Few predominant features are gained by using this branch used for the classification. A few low-level features are extracted from the image, which has the separable convolution and uses the Sobel kernel. The over-fitting problem is raised when the number of parameters is increased because of the many branches.

Further, the parameters are adjusted, which is necessary by the depth-wise separable convolution. Various stages give the features of both the separable convolution branch and the auto-encoder branch are passed and extracted to the important architecture of the classification. Multiple paths are divided from the input image with the features from the prior concatenation of two branches. DenseNet-201 [24] accepts the input image again that is the backbone of the connected layers with the sizes 4, 64, and 128 for classifying the images into 4 scores that are assimilated. The loss function is minimized by using the proper optimization. Then the loss function is used by categorical cross-entropy, and the loss is minimized using the Adam optimizer. Fig 2 depicts the suggested CNN.

3.6. Stacking LSTM

The temporal features are bear since the input consists of the series presentation of the data, such as information to recognize the data. A naïve technique is to stack the input images based on the relevant classes, and the predictions are made via the CNN to carry only. Yet, this type of technique leads to information about lost information. The units of the LSTM architecture for the recurrent neural network in the proposed system are initiated (see Fig 3) that use the memory cells for storing, modifying, accessing the internal state, and discovering the temporal data. Conventional CNN is the invariant series. On the other hand, the series of frames are considered by the recurrent neural network using the encapsulation of the real-time temporal data, which improves the complete network performance. The series of hidden vector $h = (h_1, h_2, \ldots \ldots, h_T)$ is calculated by the standard recurrent neural network and the calculate the $y = (y_1, y_2, \ldots \ldots, y_T)$ as the sequence of the output vector from the $x = (x_1, x_2, \ldots \ldots, xT)$.

$$h_t = \sigma(W_{ih}x_t + W_{hh}h_{t-1} + b_h) \quad (9)$$

$$y_t = W_{ho}h_t \mid b_0 \quad (10)$$

Here, the weight matrices are denoted by W, the bias vectors are denoted by b, and the hidden layer activation function is denoted by σ. The architecture of LSTM utilizes the memory cells for storing and output data, which allows for the better discovery of long-range temporal relationships that is greatly reliable for the dependencies in a long time. Every unit of LSTM has three gates: forget gate, output gate, and input gate.

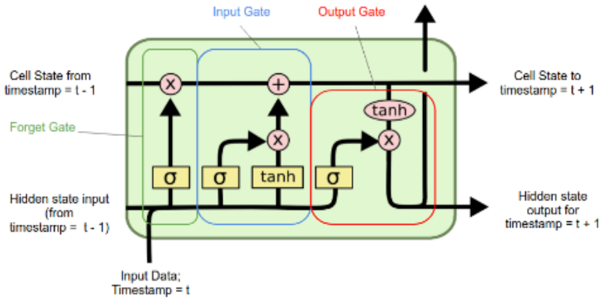

Fig 3 Single LSTM unit

Rather Fig 3 shows the single unit of LSTM. The data is captured by cells across a particular period, and the flow of information is regulated using other gates. The input gate adds the information to the cell, the unwanted information is removed by the forget gate from the cell, and the necessary information is selected by the output gate selectively from the present cell [26]. The hyperbolic tangent (tanh)d and sigmoid (σ) functions are utilized as the unit cell's activation functions. The layers of LSTM try to compute one weight parameter to optimize the three gates that combine the temporal data. The LSTM is considered in the proposed system from the suggested CNN layer, which has the output at every next frame of video. One layer of LSTM has the output as the next layer's input. The suggested CNN has the output defined, which is forwarded via the time and upwards via the stacked LSTMs, which have three layers. The softmax layer normalizes the probability vector for the four classes. The suggested weight of CNN is used for the score prediction. Moreover, the efficiency of the combined weight of CNN-LSTM for the prediction of frames depends on both temporal or sequential and spatial features having greater accuracy if the video is given that is investigated in the outcomes of the proposed system.

4. Experimental outcomes

The partitioning of the dataset is done to test set and train set in a separate way for both linear and convex probes. The approach of 5-fold cross-validation on the provided dataset is suited for achieving the test and trained data. The input obtained by frames is 20% for the linear probe data placed in the test set. On the other hand, the train set is 80% of other data for every cross-validation stage. In the same way, at every stage, 20% are placed on testing for the data of convex probe and the balance 80% are utilized for training. Every stage is used in the Adam optimizer [27], having the rate of learning $1e-3$, size of batch 64, and epochs 120 for both LSTM and CNN. After convolutional layers, many dropout layers are utilized to avoid over-fitting [28]. The loss curve and the model's accuracy are shown in Fig 9 and 10 for one of the stages of the cross-validation. The validation curves and training in the middle of the minimal gap represent the better fit in the figure at the stage of training. Table 1 gives a better understanding that the imbalance class presented in the dataset is the general situation for the medical dataset. The augmentation of data is carried out on the training data to handle the issue, which has horizontal and vertical shifts such as 0% ± 20%, horizontal (0◦ ± 360◦), and (0% ± 20%) as scaling, and vertical and horizontal flips at the stage of training and the balance across the quantity of frame is obtained in the relevant four classes.

4.1 Classification outcomes

The suggested system has the performance to demonstrate based on the three baselines. They are (i) the architecture of DenseNet-201, (ii) the suggested architecture of CNN with no block of LSTM, and (ii) the integration of the suggested architecture of CNN with the block of LSTM. The evaluation parameters are sensitivity, accuracy, F1 score, and specificity, which are concerned with three baselines. The detailed results of the -fold cross-validation following the comprehensive analysis are demonstrated below. Primarily, the dataset is used by the DenseNet-201 model having ImageNet [29] pre-trained weight for classifying the data to present the two severe scores. The better results are extracted by the DenseNet, which needs to be tuned in a better way, and the process involves the five stages of cross-validation using 20% of training and 80% of testing with the unseen data. There is 91% accuracy for the data of the convex probe and 91% for the linear probe, which is not satisfactory.

Then, the suggested architecture of CNN is established on the dataset rather than the conventional DenseNet-201. The parameters needed for evaluation are improved after establishing the suggested CNN. Then, the suggested network was combined with the blocks of LSTM and CNN to obtain a better outcome. The implementation of the suggested network is presented in Table 2 using the gradual enhancement in the outcome with the error margin presented with the level of confidence as 95%. The suggested CNN is adjoined with the block of LSTM using the improvement in the complete performance is guaranteed to have the enhancement in the accuracy 11% to 12% from the conventional architecture of DenseNet-201, and the suggested CNN gives the enhancement of 7% to 9% that maximizes the 38% to 40% accuracy at the first stage from the DenseNet-201 only. The considerable enhancement by the other parameters from the average values having the minimum deviation.

It is important to consider that the 5 fold cross-validation approach is carried out to appreciate the suggested system. The elaborated outcomes have every stage of cross-validations presented for the suggested network as LSTM + CNN in Table 3. The error margin has a confidence level of 96%, which is presented by the average values. Every validation stage has the manifestation of consistent performance, having a 91% average accuracy for the data of the convex probe. The table is evident for the complete deviation, which is inconsiderable from the average value. The perception to analyze the outcomes with the complete performance is best in the obtained images from the linear probe over the convex probe. The linear transducers are concerned with the quality of the image [30], which is the probable result of better performance in the prediction of having the images of the linear probe over the attained images of the convex probe. The consideration of the linear probes is more effective in magnifying the small consolidations even though the lung consolidations are visible in every probes sort [31].

Table 1 Parameter evaluation

Type	Model	Evaluation parameter			
		Accuracy	Sensitivity	Specificity	F1-score
Linear	DenseNet	0.57 ± 0.10	0.57 ± 0.10	0.87 ± 0.04	0.57 ± 0.13
	CNN	0.70 ± 0.09	0.70 ± 0.09	0.90 ± 0.05	0.70 ± 0.12
	CNN+LSTM	0.79 ± 0.05	0.79 ± 0.05	0.90 ± 0.03	0.78 ± 0.05
	Stacked model	0.91 ± 0.10	0.91 ± 0.03	0.90 ± 0.25	0.90 ± 0.25
Convex	DenseNet	0.53 ± 0.03	0.53 ± 0.03	0.66 ± 0.08	0.51 ± 0.07
	CNN	0.61 ± 0.04	0.61 ± 0.04	0.75 ± 0.09	0.58 ± 0.03
	CNN+LSTM	0.67 ± 0.03	0.67 ± 0.04	0.76 ± 0.14	0.66 ± 0.03
	Stacked model	0.91 ± 0.15	0.91 ± 0.03	0.90 ± 0.25	0.90 ± 0.25

Table 2 5-fold CV

Type	Model	Cross-validation					
		$\boldsymbol{K=1}$	$\boldsymbol{K=2}$	$\boldsymbol{K=3}$	$\boldsymbol{K=4}$	$\boldsymbol{K=5}$	Average
Linear	Accuracy	91.0	91.4	90.7	89.1	90.1	90 ± 0.75
	Sensitivity	89	91.4	88.7	79.1	89.1	87 ± 0.46
	Specificity	94.7	90.2	84.4	92.3	92.3	90 ± 0.78
	F1-score	89.1	92.9	89.1	77.0	88.0	87 ± 0.22
Convex	Accuracy	91.1	86.8	92.2	92.6	90.0	91 ± 0.98
	Sensitivity	71.9	86.8	91.2	67.6	90.0	81 ± 0.5
	Specificity	71.9	91.0	82.7	84.8	85.9	83 ± 0.26
	F1-score	86.6	82.8	62.2	70.5	86.5	77 ± 0.72

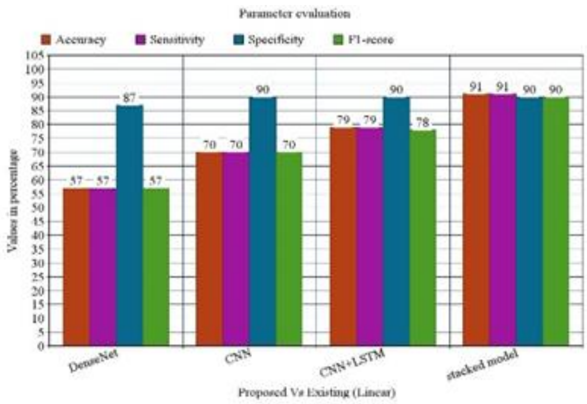

Fig 4 Linear parameter evaluation

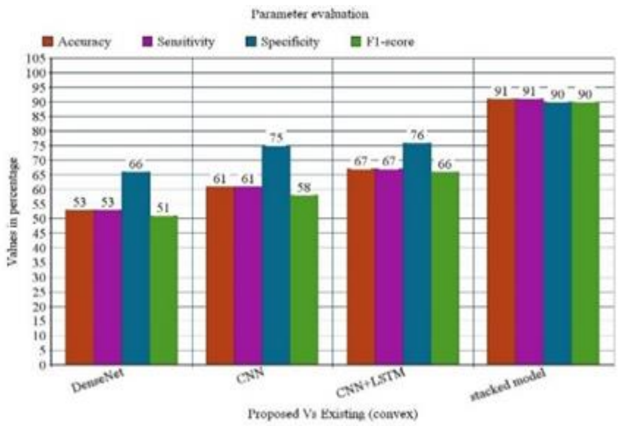

Fig 5 Convex parameter evaluation

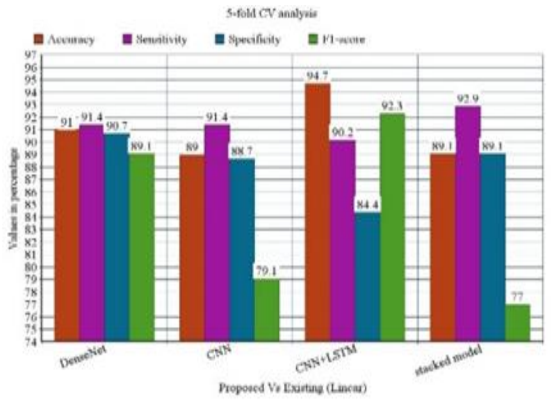

Fig 6 Linear 5-fold CV analysis

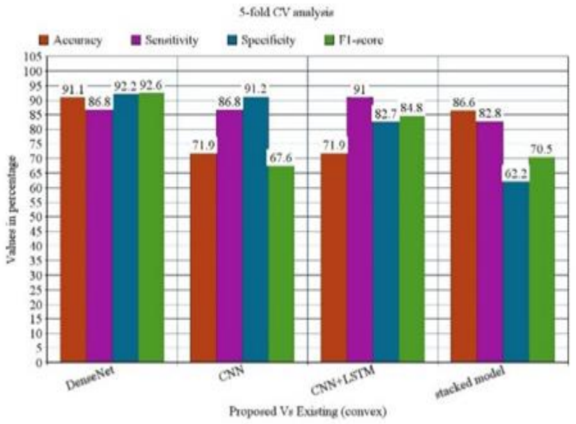

Fig 7 Convex 5-fold CV analysis

The suggested network is implemented for the gradual enhancement as the visualization presented in Fig 4 shows the data from the linear probe and another from the convex probe, which is gathered from two patients from two hospitals. The matrices indicate considerable enhancement, which is the outcome of the suggested integration model. Fig 5 has the data of linear probe with the suggested LSTM, and the CNN network can find most of the 2 frames of score in the accurate way that are unpredicted score as 1 and 0 after the establishment model of the CNN only. In score 3, the number of false negatives is lowered in the network of LSTM and CNN. In the same way, Fig 6 has a lower number of false negatives that are considered after the suggested network of LSTM and CNN. Moreover, scores 2 and 3 differ based on the consolidation magnification [14]. The CNN is implemented with the blocks of LSTM to find the subtle difference between the vectors and accurately predict the severe score.

The hospital-oriented consideration is done to check the performance of the suggested model. Consider an instance that the model is used for the gathered data with the convex probe from the patients. The suggested system is trained using the 8 patients' data, tested on the data greater than 20%, unseen from 4 patients. A better result is obtained by the suggested model having the severe enhancement in all the parameters over the prior situation, which is presented in Table 3, in which the trained model is there with no regard for the source. The average accuracy of 90% is obtained by the prediction outcomes based on data in the case of hospital oriented. On the other hand, 91% accuracy is obtained by the similar suggested network in the case of hospital oriented. The considerable enhancement has the specificity, F1 score, and sensitivity over the prior situation.

Lastly, the gradient-based class activation mapping (Grad-CAM) algorithm generates the heat maps to extract the best understanding of the visual for the targeted areas in every image [33]. The data is preserved via the different layers of suggested architecture to follow the Grad-CAM implementation to exploit and show the heat map that is the part of the input images to activate the final result in prediction. Here, Grad-CAM shows to localize the few frames presented in Fig 7. The model's activation is done in the wide area across the pleural line for the frames of score 0 in both data of linear probe and data of convex probe like the predicted identification for the affected heart that are not presented. Most data are not exhibited by Grad-CAM sometimes and always concentrate on extra regions rather than on the most related ones [7]. Further, the Grad-CAM algorithm has spread with the concentration than the compared greater area in this situation. The suggested algorithm must predict the severe scores of the provided input having less time. The average is taken, which is lower than 6 ms for performing the test having the reasonable configuration of machine for the data having the frame interval as 100 ms. Hence, there is the integration with care devices for predicting the patient's condition in real-world situations. When the complete data is provided to the algorithm, this model is available for the prediction of the complete condition of the patient. The temporal feature is concerned with the spatial feature in recorded version or real-time, and the suggested hybrid model predicts the scores of severity having greater accuracy when the data is utilized as input. Moreover, the suggested model of CNN also predicts the score for the particular frame.

4.2 Comparison

Research on the subjects based on heart disease is rarely tried to compare with other approaches. The researchers use a total sample [7] for classifying the scores of severity and claims for publishing it, yet some samples alone have been accessed in public till now. Further, directly comparing the proposed system is not acceptable in this situation. The results of classification are particular to the hospital [17]. The similar source from the data has the similarity to the sample that differed in the kind of used apparatus for the investigation by the medical practitioner to enable the models of deep learning or machine learning for the prediction to have higher accuracy. Table 3 presents one example for verifying the suggested model consistency on the same types of data that are hospital oriented in which the cases have the accuracy improved to 20% to 40% over the average independent cases of the hospital. The data relevant to the hospital is restricted considerably for training the models of DL in quantity. A trained model is used on the data rather than the source in the broader sense as the suggested one in the research, which is needed to concern the global application that is effective for predicting hospital-oriented patients. The prediction task based on the frame is independent of the hospital [7].

Table 3 Result analysis

Parameter	Dependent data			Independent data		
	CNN	CNN + LSTM	Stacked network model	CNN	CNN + LSTM	Stacked network model
Accuracy	78.5	79.2	91.2	61.0	67.7	90.1
Sensitivity	78.5	79.2	91.2	61.0	67.7	90.1
Specificity	85.7	87.2	93.5	75.6	76.8	91.5
F1-score	78.4	79	93	58.6	66.6	91.5

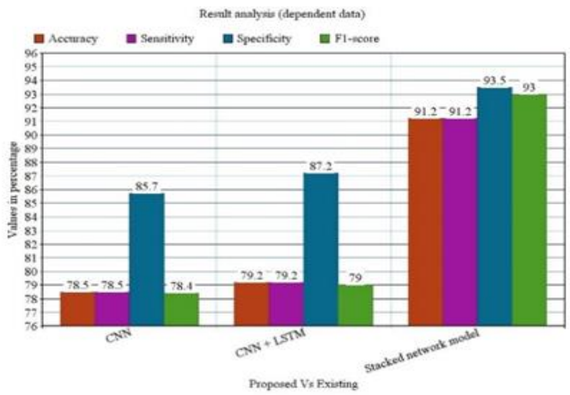

Fig 8 Result analysis of dependent data

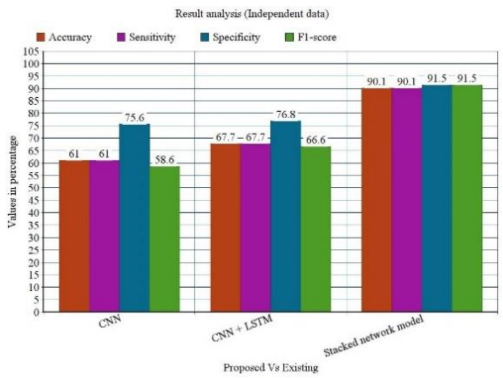

Fig 9 Result analysis of dependent data

Fig 10 Training and validation accuracy

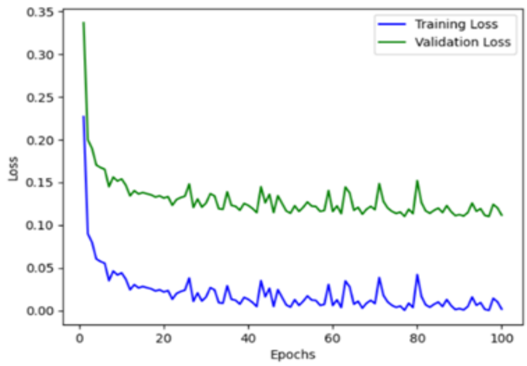

Fig 11 Training and validation loss

5. Conclusion

The integration of the recurrent neural network and convolutional model is suggested in the proposed system to predict the disease severity based on the frame for classifying the levels of outlier scores ranging from 0 to 3. The suggested block of stacked CNN introduces the embranchments with the model of DenseNet-201 having the initial branch of auto-encoder to improve the network classification performance to make sure the noise-free, powerful features, which results in a 20% to 40% boost in the accuracy of classification in terms of the model which is DenseNet-201. The LSTM layers block follows the CNN block concerning the health data in real-time to enhance the prediction accuracy using the average of 20% to 40% over the suggested model of stacked CNN only. Compared with the suggested stacked CNN, the specificity, F1 score, and sensitivity increased by 1%, 6% to 8%, and 7% to 9% accordingly that is increased by themselves, with the specificity and sensitivity by 3% to 9% and 7% to 12% over DenseNet-201.

In the hospital-oriented cases, the suggested network predicts the data source having the immense enhancement with the increase of 20% accuracy for the prediction over the cases of independent hospitals. The perception of consistent performance is done in every stage of five cross-validations having less deviation of 20% to 40% from the average value. The suggested system can predict the severity scores in less time; hence, it is established in real-world situations. Accurate comparison and evaluation are relatively having the researchers present that the suggested integrated network obtains the favourable performance to predict the severity scores. The suggested model has the performances restricted relative to the cases of the convex probe. If the data is available, the performance is improved; the only way to use more training data. However, different processing models are tested on the frames before entering the network classification. Lastly, the suggested work has one possible future job for designing the architecture based on deep learning for carrying out the upcoming procedure.

References

[1] Aggarwal, C., Zhao, Y., Yu, P.: Outlier detection in graph streams. In: Proceedings of the 27th International Conference on Data Engineering, pp. 399–409 (2011)

[2] Chen, J., Sathe, S., Aggarwal, C., Turaga, D.: Outlier detection with autoencoder ensembles. In: Proceedings of the 2017 SIAM International Conference on Data Mining, SIAM, pp. 90–98 (2017)

[3] De Stefani, J., Le Borgne, Y.A., Caelen, O., Hattab, D., Bontempi, G.: Batch and incremental dynamic factor machine learning for multivariate and multi-step-ahead forecasting. Int. J. Data Sci. Anal. (2018).

[4] Ferdowsi, H., Jagannathan, S., Zawodniok, M.: An online outlier identification and removal scheme for improving fault detection performance. IEEE Trans. Neural Netw. Learn. Syst. 25(5), 908– 919 (2014)

[5] File to, R., May, C., Renso, C., Pelekis, N., Klein, D., Theodoridis, Y.: The Baquara2 knowledge-based framework for semantic enrichment and analysis of movement data. Data Knowl. Eng. 98, 104–122 (2015)

[6] Chandarana, D. R., and M. V. Dhamecha 2015. A survey for different approaches to Outlier Detection in data mining. International Conference on Electrical, Electronics, Signals, Communication and Optimization (EESCO). Visakhapatnam, India

[7] Guo, F., C. Shi, X. Li, J. He, and W. Xi 2018. Outlier detection is based on the data structure. 2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, IEEE, 1–6.

[8] Jafarizadeh, V., A. Keshavarzi, and T. Derikvand. 2017. Efficient cluster head selection using Naïve Bayes classifier for wireless sensor networks. Wireless Networks 23 (3):779–85. doi:10.1007/s11276-015-1169-8

[9] Kiani, R., S. Mahdavi, and A. Keshavarzi. 2015. Analysis and prediction of crimes by clustering and classification. International Journal of Advanced Research in Artificial Intelligence 4 (8):11–17. doi:10.14569/ISSN.2165-4069

[10] Ko, T., J. H. Lee, H. Cho, S. Cho, W. Lee, and M. Lee. 2017. Machine learning-based anomaly detection via integration of manufacturing, inspection and after-sales service data. Industrial Management & Data Systems 117:927–45.

[11] Lin, C., Q. Zhu, S. Guo, Z. Jin, Y. R. Lin, and N. Cao. 2018. Anomaly detection in spatiotemporal data via regularized non-negative tensor analysis. Data Mining and Knowledge Discovery 32 (4):1056–73.

[12] Riahi, F., and O. Schulte 2015. Model-based outlier detection for object-relational data. Computational Intelligence, 2015 IEEE Symposium Series, Cape Town, South Africa, IEEE, 1590–98.

[13] Zhang, J., J. Cao, and X. Zhu 2012. Detecting global outliers from large distributed databases. Proceedings of the 9th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2012), Sichuan, China, IEEE, 1632–36

[14] Zhao, Y., and M. K. Hryniewicki 2018. Xgbod: Improving supervised outlier detection with unsupervised representation learning. 2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, IEEE, 1–8.

[15] Zhang, Z., J. Li, C. Manikopoulos, J. Jorgenson, and J. Ucles 2001. HIDE: A hierarchical network intrusion detection system using statistical pre-processing and neural network classification. Proc. IEEE Workshop on Information Assurance and Security, 85–90.

[16] Zhang, L., C. Liu, Y. Chen, and S. Lao 2018. Abnormal Detection Research Based on Outlier Mining. 2018 11th International Conference on Intelligent Computation Technology and Automation (ICICTA), Changsha, China, IEEE, 5–7.

[17] Yang, S., and W. Liu 2011. Anomaly detection on collective moving patterns: A hidden Markov model-based solution. 2011 IEEE International Conferences on Internet of Things, Cyber, Physical and Social Computing, Dalian, China, 291–96.

[18] Vatanen, T., M. Kuusela, E. Malmi, T. Raiko, T. Aaltonen, and Y. Nagai 2012. Semisupervised detection of collective anomalies with an application in high-energy particle physics. IJCNN, Brisbane, QLD, Australia, 1–8.

[19] Rasheed, F., Alhajj, R.: A framework for periodic outlier pattern detection in time-series sequences. IEEE Trans. Cybern. 44(5), 569–582 (2014)

[20] Ma, Y., Zhang, P., Cao, Y., Guo, L.: Parallel auto-encoder for efficient outlier detection. In: Proceedings

of the 2013 IEEE International Conference on Big Data, pp. 15–17 (2013)

[21] Tran, L., Fan, L., Shahabi, C.: Distance-based outlier detection in data streams. Proc. VLDB Endow. 9(12), 1089–1100 (2016)

[22] Xu, H., Wang, Y., Cheng, L., Wang, Y., Ma, X.: Exploring a high-quality outlying feature value set for noise-resilient outlier detection in categorical data. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, ACM, New York, NY, USA, CIKM '18, pp. 17–26. (2018).

[23] Zhang, H., Nian, K., Coleman, T.F., Li, Y.: Spectral ranking and unsupervised feature selection for point, collective, and contextual anomaly detection. Int. J. Data Sci. Anal. (2018).

[24] Wang, G., Hao, J., Ma, J., Huang, L.: A new approach to intrusion detection using artificial neural networks and fuzzy clustering. Expert Syst. Appl. 37(9), 6225–6232 (2010)

[25] Söelch, M., Bayer, J., Ludersdorfer, M., van der Smagt, P.: Variational inference for on-line anomaly detection in high-dimensional time series (2016). arXiv preprint arXiv:1602.07109

[26] Miyaguchi, K., Yamanishi, K.: Online detection of continuous changes in stochastic processes. Int. J. Data Sci. Anal. 3(3), 213– 229 (2017)

[27] Rayana, S., Akoglu, L.: Less is more: building selective anomaly ensembles. ACM Trans. Knowl. Discov. Data (TKDD) 10(4), 42 (2016)

[28] Malhotra, P., Vig, L., Shroff, G., Agarwal, P.: Long short term memory networks for anomaly detection in time series. Proc. Eur. Symp. Artif. Neural Netw. 23, 89–94 (2015)

[29] Ma, J., Sun, L., Wang, H., Zhang, Y., Aickelin, U.: Supervised anomaly detection in uncertain periodic data streams. ACM Trans. Internet Technol. 16(1), 1–20 (2016)

[30] Chauhan, S., Vig, L.: Anomaly detection in ECG time signals via deep extended short-term memory networks. In: Proceedings of IEEE International Conference on Data Science and Advanced Analytics, pp. 1–7 (2015)

[31] Kumar, L. R. ., Ashokkumar, C. ., Pandey, P. S. ., Kannaiah, S. K. ., J, B. ., & Hussan, M. I. T. . (2023). Security Enhancement in Surveillance Cloud Using Machine Learning Techniques. International Journal on Recent and Innovation Trends in Computing and Communication, 11(3s), 46–55. https://doi.org/10.17762/ijritcc.v11i3s.6154

[32] Verma, D. ., Reddy, A. ., & Thota, D. S. . (2021). Fungal and Bacteria Disease Detection Using Feature Extraction with Classification Based on Deep Learning Architectures. Research Journal of Computer Systems and Engineering, 2(2), 27:32. Retrieved from https://technicaljournals.org/RJCSE/index.php/journal/article/view/29