Feedback Based Adaptive Recurrent Neural Network for Cancer Detection using Gene Data Pattern

P. Priyadharshini, B. S. E. Zoraida

Abstract: ***Cancer detecting technology plays a vital role in the medical community. Researches have shown that patients that are affected by cancer carry same type of genetic patterns in their DNA. With this in mind, this research work concentrates on analysing gene pattern for detecting cancer using deep learning algorithms. The Feedback based Adaptive Recurrent Neural Network (FA-RNN) approach is designed to classify and analyse the gene pattern recognition. The data augmentation is done to improve the quality of the input data from COSMIC dataset which includes the detection of missing values, removing the noise present in input using multiple imputations and reducing higher base value can be done using dimensionality reduction process. After obtaining the improved dataset, the training phase begins by estimating the exact weight value of feedback layer using feedback weight loop calculation technique to lessen number of repetition during training. Moreover, the error calculation is done to evaluate the exact weight values of feedback layer used for classification. Finally the classification is done by selecting the next appropriate hidden neuron using the neuron selection activation function. The performance of the Feedback based Adaptive Recurrent Neural Network technique can be analysed using the evaluation metrics accuracy, computation time and Root Mean Square Error (RMSE) and the attained results are compared with the Recursive Neural Network(RNN) and Convolutional Neural Network(CNN) algorithms. The obtained results such as higher accuracy, reduced RMSE and less computation time in Feedback based Adaptive Recurrent Neural Network indicates that it performs the enhanced operation than CNN and RNN.***

Keywords: ***Convolutional Neural Network(CNN), Recursive Neural Network (RNN), Feedback based Adaptive Recurrent Neural Network (FA-RNN).***

I. INTRODUCTION

The use of deep learning is to recognise and classify data is increasing in recent days with different machine learning algorithms. The main advantage of deep learning over any other conventional classification algorithm is their ability to adapt to the type of input and learn from each input. This type of algorithms is used widely to recognise the input data based on its significance and to produce the desired output. With each different kind of input deep learning network will learn from each pattern in input. Therefore it increases the accuracy of the network with time. However there are still some areas in deep learning that needs be improved. When the amount of available data is too low and they are highly independent from one another. In such case of less number of input data the classification is hard for neural networks. Therefore multiple hidden layer architecture for this kind of low number input data is not accurate. Hence, a complete analysis of different type of deep learning neural networks and their corresponding working in case of less number of input data need to be done. There are three types of neural network that are taken into consideration. Conventional Neural network (CNN), Recursive Neural Network (RNN) and Recurrent Neural Network (RNN).

Conventional Neural Networks (CNN) uses multilayer perceptron by connecting each node with nodes of the next hidden layer. This kind of neural network has many connections which increase the number of weights and processing time for each iteration. The processing time of algorithm is heavily dependent on the output convergences. When the number of internal connection increases over fitting occurs. This over fitting leads to increase in training and processing time which may not be suitable for real time applications. In CNN the number of hidden layer neuron is directly proportional to number of input variables. Each hidden layer of neural network will be used to classify a specific part of the input data. Therefore using a single hidden layer pattern recognition system will reduce the accuracy of CNN [1].

Recursive Neural Network (RNN) uses repetitive analysis to find the pattern in the data. In recursive neural network the weight between each node will remain unchanged until convergence. The neural network will be formed on the basis of tree structure and every data that is related to the next data will be connected to the same parent node. The hidden neurons are divided into three nodes activation node, parent node, sibling node. The activation function is used to calculate the weight of the parent node for each data type. These parent nodes are then connected to output nodes to find the output probability. This kind of approach will be suitable if the number of output and input nodes are two small. This is due to the fact that with increase in number of output nodes the probability value will be scattered and will reduce the time for convergence. This kind of algorithm is best suitable for emotion detection in word processing where the output values either positive or negative [2].Recurrent Neural Network are defined as a connectionist model containing a self-connected hidden layer.

Revised Manuscript Received on December 30, 2019.
* Correspondence Author

P. Priyadharshini*, Research Scholar, Department of Computer Science and Engineering, Bharathidasan University, Tiruchirappalli, India. Email: priya.tanusri@gmail.com

Dr. B. S. E. Zoraida, Assistant Professor, Department of Computer Science and Engineering, Bharathidasan University, Tiruchirappalli, India. Email: b.s.e.zoraida@gmail.com

The main benefit of the recurrent neural network is that a memory of previous inputs remains in the network internal memory allowing it to make use of past results to compare it with present states. This kind of memory comparison will be suitable for data pattern recognition.

Despite of this many advantages there are few other disadvantages such as vanishing gradient. This can be rectified with the introduction of long term and short term memory functions into the algorithm. Increase in data complexity increases the need of more hidden layers. Therefore, the accuracy of the network is directly proportional to the number of hidden layers in the network. But with increase in number of Hidden layers increases the computational cost and decreases the neural network efficiency. Therefore an efficient neural network with reduced number of hidden layers and increased performance need to be proposed [3].

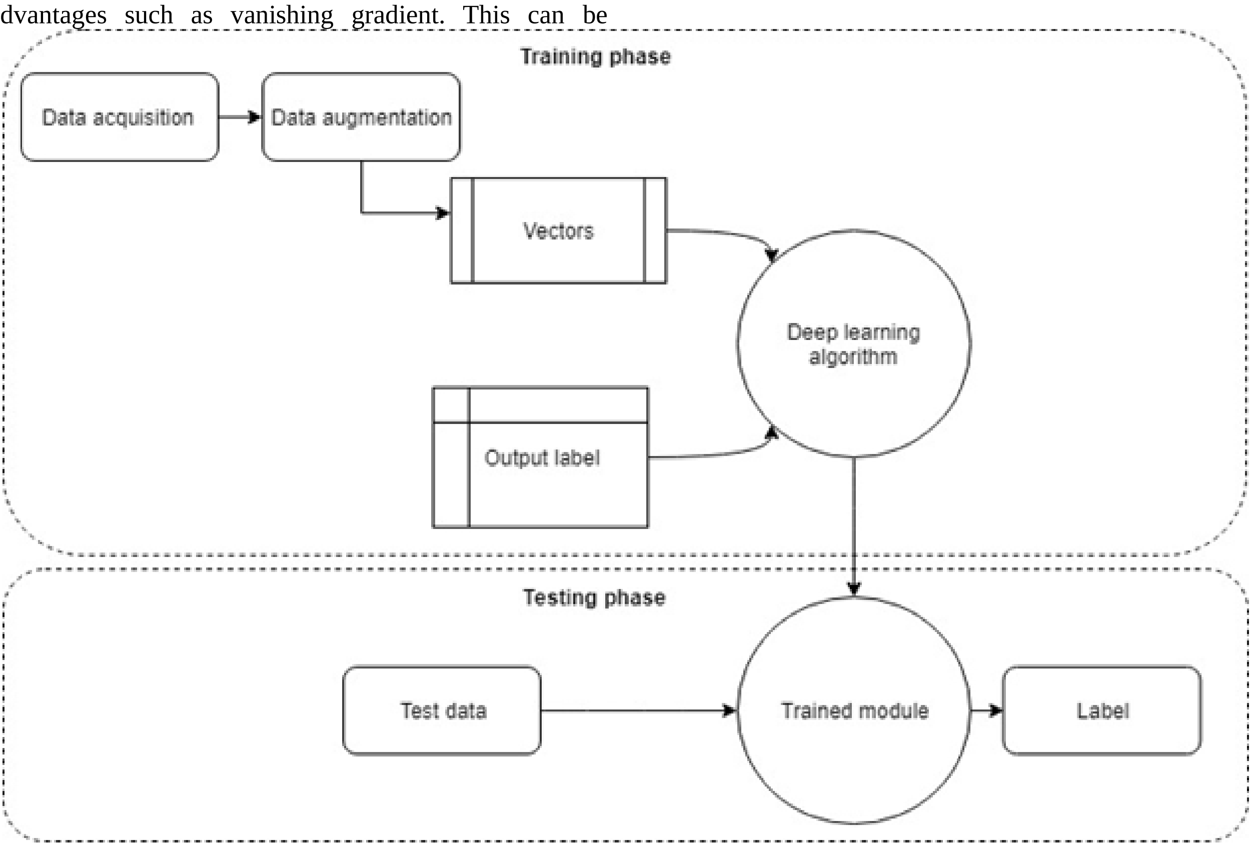


Fig. 1: Basic functional diagram of Recurrent neural network

The data consists of two clauses data IDs, gene mutation data. The weightage for each of the data in neural network is given based on the type of data and influence of that data on output [4] The total data classification is done in following steps. 1) Data acquisition 2) Data augmentation 3) Data separation 4) Training and 5) Testing. These are the steps normally taken for any kind of neural networks. The simulation of the algorithm FA-RNN, CNN and RNN are carried out in MATLAB environment. The database for our implementation is acquired from COSMIC cancer mutation database [5]. The FA-RNN experimental results were compared with Convolutional Neural Network (CNN) and recursive neural networks (RNN).

The basic functional diagram for deep learning neural network is shown in above Fig. 1. The first part of data processing is that data is collected from database and data is augmented with the help of numerosity reduction technique. The augmented data is then processed with the help of deep learning algorithm. The trained values are then stored in a training module. These trained values are then compared in future for cancer classification.

The following work is summarized as follows, Section II discusses the literature review, section III describes the methodology part and algorithm is discussed in section IV, Results and discussion is mentioned in Section V and conclusion is stated in section VI.

II. LITERATURE REVIEW

Cancer detection using pattern recognition is most common topic in recent days due to the increased availability of data through cloud. MF Mohammed and CP Lim et al (2014) proposed a pattern recognition algorithm based on fuzzy max-min technique. The main objective of this paper is to find perfect pattern recognition for their online learning platform. The data pattern recognition done in this paper is able to perform with both normal and medical datasets [6].

Hideaki Hayashi et al (2015) proposed data are classified with the help of recurrent probabilistic neural network on basis of time-series discriminant component analysis. The model involves compression of high dimensional data to lower dimensional data . The results of both dimensionality and classification are obtained at end of neural network [7]. Uma Ojha et al (2017) proposed prediction of cancer using data mining technique is famous in recent years predicted the reoccurrence of cancer in patients based on some standard parameters. There are more than 32 numbers of attributes that are taken from the patients with one outcome value.

The parameters that are collected from the patients are analysed with the help of classification algorithms such as SVM and decision tree algorithms. The results are then compared with clustering algorithms on the basis of accuracy, sensitivity and specificity. With their data mining based approach of cancer detection the accuracy value of classification algorithm is 81% when compared to fuzzy means clustering algorithm with accuracy of about 37% [8]. Yawen Xiao et al (2018) proposed Cancer detection using genetic data is comparatively hard because of the fact that there are so many factors in gene that can contribute to cancer. Even after prediction of cancer there is still some chance for error due to external factors.

In spite of this drawbacks auto encoder based cancer prediction method using RNA sequence data. In this research they used semi supervised deep learning method to predict cancer using RNA sequence data such as gene length, mutation values etc. the proposed auto encoder based deep learning algorithm uses greedy layer-wise pre-training and scarcity penalty values to extract important information from high dimensional data. The data obtained with these methods are then classified with the help of deep learning algorithms [9]. Shraddha Deshmukh and Swati Shinde (2016) proposed the main concept to find cancer in patients using cancer datasets. The real data such as length, gene score, mutation score which are available after clinical examination are classified with the help of neural network. These data are classified with different types of lung diseases with the help of fuzzy min-max approach. The dataset from various lung diseases were collected and these values are analysed for patterns to classify the cancer in patents. This kind of simple pattern recognition takes smaller steps. Therefore, the time and efficiency of this algorithm also increases. The only thing that separates it is its lack of learning capability [10]. Xiaohui Yuan et al (2018) proposed deep learning algorithm to classify the bowel cancer in patients using multi-class data.

The main use of this algorithm is when the number of positive case data from patients is lower than negative cases. This type of classification will reduce the accuracy of classifiers. Therefore, they have introduced a self-penalizing technique to increase the value if the classifier produces a wrong classification suing training. This kind of approach will increase the efficiency of classifiers [11]. FengWang et al (2019) proposed data pattern recognition in blood pressure records using convolutional neural network. They first separated multiple session patterns of intradialytic blood pressure, then analysed Blood pressure in patients via systematic approaches. They separated each input to convolutional kernels, by using average pooling the overall response of system also increases. This type of two way strategies solves data imbalance in the network [12]. MatthieuScherpf et al (2019) proposed use of recurrent neural network for pattern recognition is quite common. The authors proposed recurrent neural network to predict Sepsis from MIMIC III database.

By examining the sequence length of patient data in deep learning algorithm and using receiver operating characteristic (AUROC) for performing classification task. They improved the classification accuracy when compared to other heuristic algorithms [13].

III. METHODOLOGY

The input data are acquired from COSMIC database for recognising pattern. The parameters that are taken into consideration are HGNC ID, Sample ID, tumour ID, mutation ID, Mutation genome position, Gene CDS length, Mutation genome position score, Mutation strand, FATHMM score. To make this data easy to process, each data will be converted into similar pattern with the help of data augmentation algorithm.

The basic building block of Convolutional Neural Network is its multilayer perception principle. Multi-layer perception is when each neuron in the network is connected with every other neuron in the network. To find the accuracy and time consumption the hidden layers will be increased from 1 to 3. Sigmoid function is used as an activation function.

Recursive Neural Network works by applying same set of weight over and over again until a convergence.

To develop Recursive Neural Network tanh function is used as activation function and number of nodes is increased from 1000 to 8000 and hidden layer is increased from 1 to 5.

The basic structure of Feedback based Adaptive Recurrent Neural Network (FA-RNN) is shown Fig 2.

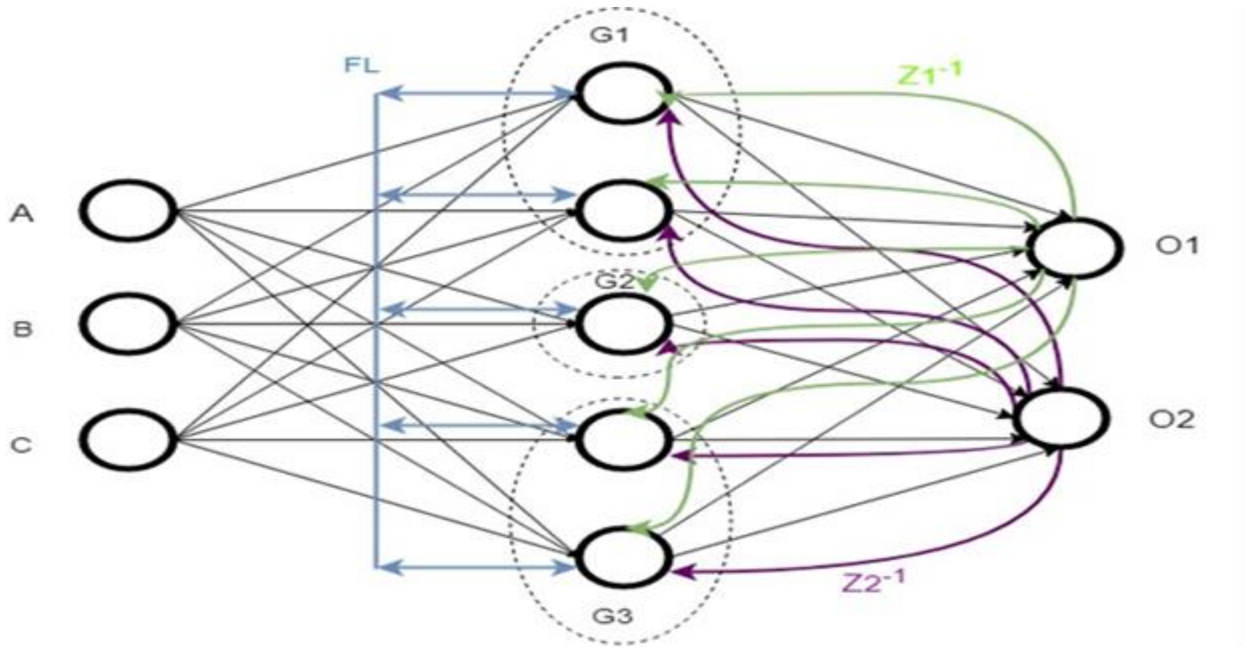


Fig. 2: FA-RNN Architecture

In this Fig. 2. A,B,C represents input nodes. G1, G2 and G3 represent the groups or clauses that are formed based the type of data input. O1 and O2 represent the output in the network. Z_1^{-1} and Z_2^{-1}signifies feedback gain from the output nodes. The basic construction of FA-RNN consists of single input layer and single hidden layer, the reason to do that is to reduce the latency in algorithm computation.

The main function that separates FA-RNN is feedback circuit both from hidden layers and output layers. This measure of back-propagation and next neuron value from feedback loop increases the classification efficiency. The input to G1node in hidden layer will be shown in (1).

The transfer function from feedback loop between each node in the neural network. G is the gain from node and n represents nodes in hidden layer shown in (2).

$$G1 = f(\varsigma)\begin{pmatrix} Aw_1 + Bw_1 + Cw_1 + FL_1 + (\frac{1}{z_{\ 1}}(O1\times G1))... \\ +(\frac{1}{z_{\ 2}}(O2\times G1)) \end{pmatrix} \tag{1}$$

$$n_i = \frac{G_i(S)}{1+G_i(S)G_{i-1}(S)} n_{i+1} \tag{2}$$

The function used in this work is sigmoid function. This function is used to find the output is 0 or 1.

The neural network consists of two feedback layers one within the hidden layer and one between layers. The steps for FA-RNN algorithm is broken down into four parts Data augmentation, Weight calculation, Error calculation and Data classification.

A.Data augmentation

The data augmentation is carried out in three stages missing value estimation, data noise cancellation and dimensionality reduction [14].

a. Missing value estimation

The databases that are fetched from patients may contain one or two missing values. This kind of missing values may be lethal which may harm the performance of neural network. To fill these missing values from the dataset, mean values of maximum and minimum numbers are taken and these values are replaced in empty data as expressed in following (3).

$$E = \mu(\max(x), \min(x)) \tag{3}$$

Where, max(x), min(x) is the maximum and minimum value from missing vector in the database. μ is the mean value of maximum and minimum and E is the output of missing value.

b. Data noise cancellation

The data noise from genetic database represents any irrelevant data that are present in the input data. To remove this irrelevant data from the database we used multiple imputation method. In multiple imputations, a data with high random distribution will be considered as noisy and that will be replaced with the help of Bayesian posterior distribution principle.

The formula for multiple imputations using Bayesian distribution is given in below (4).

$$X_{noise} = \mu_{noise} + (\mathrm{x}_{obs} - \mu_{obs})\mathrm{B} + \mathrm{e} \tag{4}$$

Where, X_{noise} is the new output value of noisy input, μ_{noise} is the total mean value of the vector, x_{obs} is the absolute value of the input and μ_{obs} is mean value of the absolute, B represents the Bayesian distribution function and 'e' is the normal distribution value.

c. Dimensionality reduction

There are many data in the database that consists of large base values.

This base values are converted into smaller base value with the help of Principle Component Analysis (PCA) function. In this function the value with higher base will be converted into its Eigen value and Eigen vector demonstrated in (5).

$$[V, D] = \lambda(X) \tag{5}$$

Where, V is the eigen vector and D is the eigen value of input number X.

$$X_{out} = V + base(D).D \tag{6}$$

The value Xout derived in (6) will be in integer form with Eigen value as decimal and Eigen vector value as real number. This kind of dimensionality reduction will reduce the complexity of neural network weight estimation function.

B. Weight calculation

Weight is the connection between different nodes and layers. This weight is used to assign the path of classification algorithm.

In FA-RNN algorithm there are two different weight values to be calculated, first weight will be between nodes of different layers and second will be between individual nodes in hidden layer (FL) [15]. Before training the neural network, initial weight between each connection is assigned randomly in the range of 0 to 1.

The weight value will be adjusted during each training iterations based on proposed priority based weight calculations function expressed in (7).

$$W_O = W_I \pm (\varepsilon \times n(FL)) \tag{7}$$

Where, w_o is the output weight value of individual connection,w_I is the weight of previous training iteration, ε is the error value from output, *n(FL)* is the number in FL (Feedback Loop) table.

a. Feedback Loop weight calculation (FL)

The value of feedback layer can be calculated based on the highest weight value in previous training phase. In each training cycle the value of weights will be altered and this value will be stored in a sorting order in feedback loop as shown in Fig. 3. For example, if the weight value of node 1 is greater than the weight value of node 2 by 20% then FL1 will be increased to 0.2 and value for FL2 will be 0.

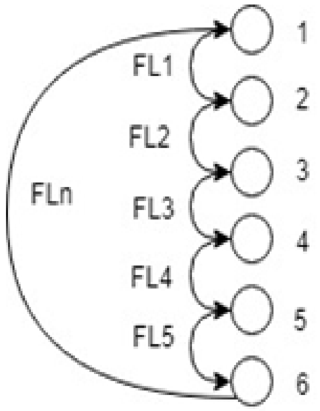


Fig. 3: Basic feedback loop connection

The value of feedback loop will be adjusted until last iterations. The nodes that have very low weight changes in feedback loop will be assigned as zero. This makes the node unable to participate in the next iteration. This kind of dropout technique reduces the number of cycles and computational time in the network. This kind of internal feedback structures will help to reduce the number of iterations for training and reduce overfitting [16].

C. Error calculation

The value of error is the measure of difference between predicted value and desired value, the value will be altered based on the value of output.

$$e = \frac{1}{2N}\sum_{i=1}^{N}(\overline{x_i} - x_i) \tag{8}$$

In this above (8), the value of $\overline{x_i}$ is the desired value, x_i is the present output value, N is the number of neurons. This error value will be a floating point ranging from 0 to 1.The value of neurons, weight and feedback loop values will be evaluated until the final convergence is achieved. The evaluated values will be stored in a dataset to classify the input data.

D. Data classification

In classification phase the input data will be converted based on the data augmentation technique. Then the value of input neurons will be formulated with weight values and the resultant values will be calculated. With the help of feedback loop the hidden neuron that is most likely to be selected will be calculated with the help of neuron selection model.

a. Neuron selection

In this phase the next neuron value will be selected or calculated based on value of activation function. Activation function is a mathematical formula which determines neuron value and next neuron to be activated during classification.

This kind of neuron selection will eliminate the unwanted time taken for processing other neurons during testing. The type of neuron that is selected will be based upon the value of activation function using expression mentioned in (9).

$$S = \frac{1}{1 + e^{-k(ax)}} \tag{9}$$

Where, k is the steepness coefficient which determines the variation in x-coefficient values.

Higher the value of k higher the value of x, ‘a’ is the value of total gain of the neuron.

The final value of x is the value of input neurons ‘G1’ as shown in (1).With this principle the output value of given input can be found by analysing the values of hidden neurons and their respective weight values.

IV. ALGORITHM

FA-RNN (TRAINING)
Input: ID, genetic data
Output: Trained neural network

Step 1: The dataset that need to be trained are converted into similar data format using unified data augmentation.

Step 2: For ‘n’ number of input neurons 2^n-1 number of hidden neurons will be assigned.

Step 3: The value of weight between nodes will be assigned randomly.
Step 4: The value of each node will be calculated using (1).

$$G1 = f(\varsigma)\left(\begin{array}{l} Aw_1 + Bw_1 + Cw_1 + FL_1 + (\frac{1}{z_1}(O1 \times G1)) + ... \\ (\frac{1}{z_2}(O2 \times G1)) \end{array}\right)$$

Step 5: The node values will be multiplied with its respective weights and the predicted output result will be calculated.
Step 6: This predicted result will be compared with desired result to find the error value.
Step 7: This error value will be used to update the weight during each iterations. From (7),

$$W_O = W_I \pm (\varepsilon \times n(FL))$$

Step 8: The position or value of each feedback loop will be updated using error value.

Step 9: This position value is used to eliminate the nodes that have low weight differences. Therefore, this dropout quality can decrease the computational time.

Step 10: The iterations will be carried out till convergence.

Step 11: The calculated value will be stored in a dataset.

FA-RNN (TESTING)
Input: training dataset, genetic data of particular person.
Output: Classification output.

Step 1: Input data will be converted into correct format.
Step 2: The node value will be calculated with the help of same formula.
Step 3: Using activation function the next node or matching hidden node will be calculated using (9).

$$S = \frac{1}{1 + e^{-k(ax)}}$$

Step 4: Then the value of hidden node will be multiplied with the weight values.
Step 5: The node with higher value will be selected as the output.
Step 6: Stop.

V. RESULTS AND DISCUSSION

The performance metrics where analysed by comparing it with deep learning algorithms such as FA-RNN, CNN and RNN. The simulation of this three algorithms where carried out in MATLAB R2016 and run on a system with 2.6 GHz clock and 8 GB RAM, in a Microsoft Windows 8.0 environment. The data for analysis are taken from COSMIC dataset, in this dataset all the unwanted data are removed and only nine data where selected as the input data. Those data include HGNC ID, sample ID, tumour ID, mutation ID, Mutation genome position, Gene CDS length, Mutation genome position score, Mutation strand, and FATHMM score.

To evaluate the performance of neural network metrics such as accuracy, Root Mean Square Error (RMSE), time consumption [17].

a. Accuracy

Accuracy is the measure of ability to classify the data at given time. All the data that are feed are analysed and the value of true positive, true negative are calculated.

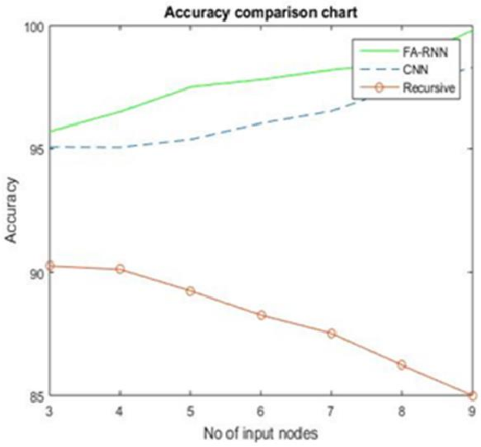


Fig. 4:Accuracy comparison plot.

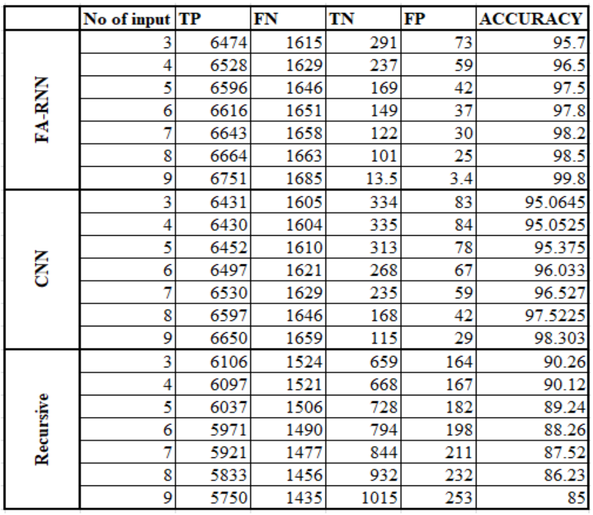
Through this true positive (TP) and false negative (FN) values the value of accuracy can be calculated using formula stated in (10).

$$Accuracy = \frac{TP + FN}{number.of.samples} \tag{10}$$

The accuracy values of FA-RNN are compared with CNN, RNN algorithm by changing the number of inputs as shown in Table 1.

From the Fig. 4, it is clear that the accuracy value of FA-RNN is high and increases with increase in inputs. This is due to the use of internal feedback loops, which reduces over fitting in the network. The accuracy value will be increased upto 10 to 20% for each increase in inputs which shows the input handling capacity of the FA-RNN.

Table 1: Neural network performance table.



	No of input	TP	FN	TN	FP	ACCURACY
FA-RNN	3	6474	1615	291	73	95.7
	4	6528	1629	237	59	96.5
	5	6596	1646	169	42	97.5
	6	6616	1651	149	37	97.8
	7	6643	1658	122	30	98.2
	8	6664	1663	101	25	98.5
	9	6751	1685	13.5	3.4	99.8
CNN	3	6431	1605	334	83	95.0645
	4	6430	1604	335	84	95.0525
	5	6452	1610	313	78	95.375
	6	6497	1621	268	67	96.033
	7	6530	1629	235	59	96.527
	8	6597	1646	168	42	97.5225
	9	6650	1659	115	29	98.303
Recursive	3	6106	1524	659	164	90.26
	4	6097	1521	668	167	90.12
	5	6037	1506	728	182	89.24
	6	5971	1490	794	198	88.26
	7	5921	1477	844	211	87.52
	8	5833	1456	932	232	86.23
	9	5750	1435	1015	253	85

b. Root Mean Square Error (RMSE)

Root mean square value is the measure of randomness or consistency of classifier by calculating the square root mean of error values. Error values are the difference between predicted value and actual value. The formula used to find RMSE is expressed in (11).

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\bar{x}-x\right)^{2}} \qquad (11)$$

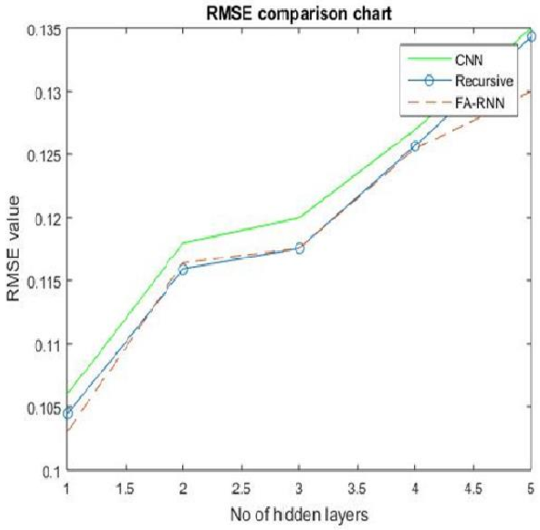


Fig. 5: Root Mean Square value comparison.

In the above eqn(11), $\bar{x}$ the actual value and x is the predicted value. To find the RMSE value the dataset is created as input and the data are feed to the neural network.

The value of predicted output value will be analysed and compared with actual value data. This comparison of RMSE is done for FA-RNN, CNN and RNN as shown in Fig. 5.The values of RMSE for different number of hidden layers where compared, it is clear from the Fig.5, that RMSE value is comparability low in FA-RNN algorithm.

c. Computation time

The main disadvantage with other neural network is that due to increased number of iterations the computational time of neural network also increases.

The total run time can be calculated using formula which is expressed in (12).

$$T = t(n^{k}) \qquad (12)$$

Where t is the time for single iteration, n is the number of nodes, k is the number of hidden layers. Total time taken is calculated for each algorithm and plotted as shown in Fig.6.

From the Fig.6, it is clear that the value of computational time reduce with increase in number of nodes because we have only one hidden layer.

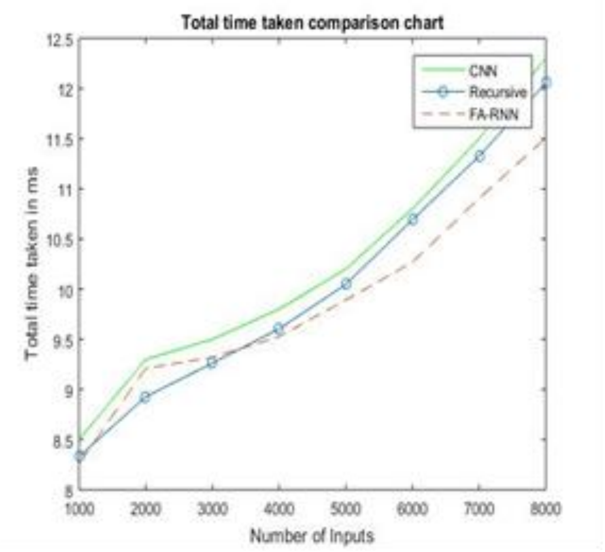


Fig. 6:Computational time comparison chart.

VI. CONCLUSION

In this research work, the deep learning algorithms like FA-RNN, CNN and RNN is implemented using COSMIC dataset. Only nine data is selected from these input dataset for providing accurate gene pattern recognition. The enhanced FA-RNN neural network may perform the efficient identification using the new feedback loops and dropout techniques. This feedback loop can bring the advantages of lessening the number of processing cycle to achieve convergence and dropout technique is introduced to eliminate overfitting in FA-RNN. The FA-RNN result shows an increase in classification accuracy with low error rate and computational time when compared to CNN and RNN. In future a hybrid model can be developed which works by analysing both genetic and image patterns for better cancer detection accuracy.

REFERENCES

1. F. F.Ting, Y. J.Tan, & K. S. Sim, “Convolutional neural network improvement for breast cancer classification”, *Expert Systems with Applications*, Vol.120, 2019, pp.103-115.
2. G. Ditzler, R.Polikar, &G. Rosen,. “Multi-layer and recursive neural networks for metagenomic classification”. *IEEE transactions on nanobioscience*, Vol. 14(6), 2015, pp.608-616.
3. M.Mohammadi, Y. H. Tan, W. Hofman, & S.H. Mousavi,.” A novel one-layer recurrent neural network for the l1-regularized least square problem”. *Neurocomputing*, Vol.315, 2018, pp. 135-144.
4. H. K..Poon, W. S Yap, Y. K..Tee, W. K. Lee, & B.M.Goi,” Hierarchical gated recurrent neural network with adversarial and virtual adversarial training on text classification”. *Neural Networks*, Vol.119,2019,pp. 299-312.
5. S. A.Forbes, N.Bindal, S.Bamford, C.Cole, C.Y. Kok,D.Beare ... & J.W.Teague,” COSMIC: mining complete cancer genomes in the Catalogue of Somatic Mutations in Cancer”, *Nucleic acids research,* Vol.39(1), 2019,pp.945-950.
6. M.F.Mohammed, &C.P. Lim, “An enhanced fuzzy min–max neural network for pattern classification”.,*IEEE transactions on neural networks and learning systems*, Vol.26(3), 2014, pp. 417-429.
7. H. Hayashi, T.Shibanoki,K. Shima,Y. Kurita, & T.Tsuji “A recurrent probabilistic neural network with dimensionality reduction based on time-series discriminant component analysis”, *IEEE transactions on neural networks and learning systems,* Vol.26(12), 2015, pp.3021-3033.

8. U.Ojha, , & S.Goel, S,” A study on prediction of breast cancer recurrence using data mining techniques”, *IEEE, 7th International Conference on Cloud Computing, Data Science & Engineering-Confluence*, 2017, pp. 527-530.
9. Y.Xiao, J.Wu, Z.Lin, , & X.Zhao,”A semi-supervised deep learning method based on stacked sparse auto-encoder for cancer prediction using RNA-seq data”, *Computer methods and Programs in Biomedicine,* Vol. 166, 2017, pp.99-105.
10. S.Deshmukh, & S.Shinde ,”Diagnosis of lung cancer using pruned fuzzy min-max neural network” ,*IEEE, International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT),* 2016, pp. 398-402.
11. X.Yuan,L. Xie, & M. Abouelenien, “A regularized ensemble framework of deep learning for cancer detection from multi-class, imbalanced training data”, *Pattern Recognition*, Vol.77, 2018,pp. 160-172.
12. F.Wang, Y.Wang,, Y.Tian, P.Zhang, J.Chen, & J.Li,“Pattern Recognition and Prognostic Analysis of Longitudinal Blood Pressure Records in Hemodialysis Treatment Based on a Convolutional Neural Network”, *Journal of biomedical informatics,*Vol.98, 2019,p.103271.
13. M.Scherpf, F.Gräßer,H. Malberg, & S.Zaunseder,” Predicting sepsis with a recurrent neural network using the MIMIC III database",*Computers in Biology and Medicine*, Vol.113,2019,p.103395.
14. S. García,J. Luengo, & F.Herrera, “Tutorial on practical tips of the most influential data preprocessing algorithms in data mining”, *Knowledge-Based Systems*, Vol.98,2016,pp. 1-29.
15. P.A.Henríquez, &G.A. Ruz, “A non-iterative method for pruning hidden neurons in neural networks with random weights”,*Applied Soft Computing*, Vol.70, 2018,pp.1109-1121.
16. W.Wang, J.Chen, T.Hong, & N.Zhu,”Occupancy prediction through Markov based feedback recurrent neural network (M-FRNN) algorithm with WiFi probe technology”, *Building and Environment,* Vol.138, 2018, pp. 160-170.
17. C.Shahnaz, J.Hossain, S.A.Fattah, S.Ghosh, & A.I. Khan,” Efficient approaches for accuracy improvement of breast cancer classification using wisconsin database”,*IEEE Region 10 Humanitarian Technology Conference (R10-HTC),* 2017, pp. 792-797.

AUTHORS PROFILE

P.Priyadharshini is currently pursuing Ph.D in Department of Computer Science and Engineering, Bharathidasan University, Tiruchirappalli, India. She had completed her Bachelor of Science in Physics from Shrimathi Indira Gandhi College from Bharathidasan University, Trichy, Tamilnadu. She had completed her M.C.A from Bharathidasan University, Trichy, Tamilnadu. She had completed her M.Phil from Bharathidasan University, Trichy, Tamilnadu. Her area of research includes Deep Learning, Image Processing and Big Data.

Dr. B.S.E.Zoraida is currently working as an Assistant Professor in the Department of Computer Science and Engineering, Bharathidasan University, Tiruchirappalli, India. She had completed her Ph.D. in Computer Science from National Institute of Technology, Trichy (NITT), Trichy, Tamilnadu. She has academic teaching experience of more than 20 years, with the publication of more than 40 papers in the reputed international journals and two chapters in Springer book series. Also, she has been taking part as a reviewer in international reputed journals. Her area of research includes computer applications in smart grid, semantic web, soft computing, and molecular computing.