Proceedings of the Third International Conference on Computing Methodologies and Communication (ICCMC 2019)
IEEE Xplore Part Number: CFP19K25-ART; ISBN: 978-1-5386-7808-4

GRU Based Deep Learning Model for Prognosis Prediction of Disease Progression

M.Pavithra
Department of CSE
Pondicherry Engineering College
Pondicherry
pavimuthu27@pec.edu

Dr K.Saruladha
Department of CSE
Pondicherry Engineering College
Pondicherry
charuladha@pec.edu

K.Sathyabama
Department of CSE
Pondicherry Engineering College
Pondicherry
Sathii_manju@pec.edu

***Abstract* —Recently different Deep Learning (DL) models have been emerging to predict the disease amelioration. Generally deep learning is the state of art for learning the multiple representation of neuron. Discriminant DL model includes different architecture and initial architecture is Recurrent Neural Network (RNN) which learn the data labeled in the form of sequence and it has long term dependency and vanishing gradient problem. Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU) is an improved version of RNN which deal with problems in RNN. Deep care LSTM is one of the model in LSTM developed by [Trang Pham et.al] for predicting diabetes disease and compared the results with markovian and support vector machine model. In deep LSTM model some problems are reported like short term trajectories, less accuracy. To overcome the problem Deep Care GRU has been proposed to identify the diabetes disease amelioration.**

Keywords— Recurrent Neural Network(RNN),long Short term Memory(LSTM),Gated Recurrent Unit(GRU) , Deep care GRU.

I. INTRODUCTION

Artificial intelligence aim to make an expert system i.e. the computer can be designed to mimic the activities of the human brain. Machine learning is a branch of artificial intelligence which allows the machine to learning by training through the past experience like human brain and performs the different tasks. The machine learning is classified into two techniques. One is unsupervised learning and other is supervised learning. The supervised learning is based on the labeled data that is it contains both input and output data. Classification and regression comes under the supervised learning which is based on some training data the model will learn and classify the results in testing. Unsupervised learning is based on the unlabeled data where it contains only input data [1]. Based on the input data the model will able to learn and make the clustering i.e. it groups the data having similar characters. And also many types of techniques are available in machine learning like semi supervised learning and reinforced learning. Deep learning is a sub field of machine learning which will make the system to learn intelligently. Machine learning algorithm requires manual work to sort error. But in deep learning the machine itself solves the problem. Deep learning model has two types of architecture discriminant and generative and it can be view as supervised and unsupervised deep learning model respectively. Discriminant model contains Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU). And generative model contains Restricted Boltzmann Machine (RBM), Deep Belief Network (DBN), and Deep Machine (DBM). CNN is used to process the two dimensional data like image and text. RNN, LSTM, GRU are used to process the text data. Generative model can be used to process dimensionality reduction and feature learning [2].

II. RELATED WORK

Predicting illness progression of the patient using the deep learning model was designed using the many deep learning models. The illness progression of the individual patient subject can be predicted based on the patient's past history. The neural network model learns the illness status of the patients from their past information and predicts the progression of the diseases. Care LSTM model was used for learning and prediction. And in care LSTM the learning was done by intervention and diagnosis code like ICHI codes and ICD 10 codes. Diabetes and mental health dataset are consider as a sample dataset in the deep care model and predict the progression of the disease along with the intervention code. And compare the prediction and intervention results of diabetes and mental health [5]. This paper predicts the readmission of the patients using the deep learning models. Hospital readmission was avoidable because sometimes it causes excessive readmission of patients. In order to avoid the excessive readmission contextual modeling [6]. Diabetes disease progression was researched using the machine learning and data mining model. It compares the performance of machine learning algorithm and data mining models. The data mining algorithm like support vector machine, k-means and k-nearest algorithm are used. And machine learning model like

CNN and RNN can be used to diabetes prediction [7].Electronic medical record contains the history of patients like blood test report, blood pressure rate etc. The EMR contains data's in different format like temporal, mixed and multimodal. These data's are derived using restricted boltzmann machine with less human supervision. It derives the data in dimensional vector space [9] Like the above approach many deep learning models are used in healthcare prediction. But still the results of above deep learning model leads to vanishing gradients and exploding gradients problem [10][11]

III. DISCRIMINATIVE MODEL

A. *Recurrent Neural Network (RNN)*

Recurrent Neural Network is a discriminative deep learning model, here the input and output of the network flows using the loop. It is the looped network and also an advanced version of Feed Forward Neural Network (FFN). FNN does not suit for time and recurrence data whereas RNN is suitable for recurrent data [3]. But RNN does not deal with long term dependencies and vanishing gradient problem. RNN makes the gradients towards zero while improving the input this makes the long term dependencies. And the long term dependencies is said to be the vanishing gradient problem. The architecture model of RNN is illustrated in figure 1

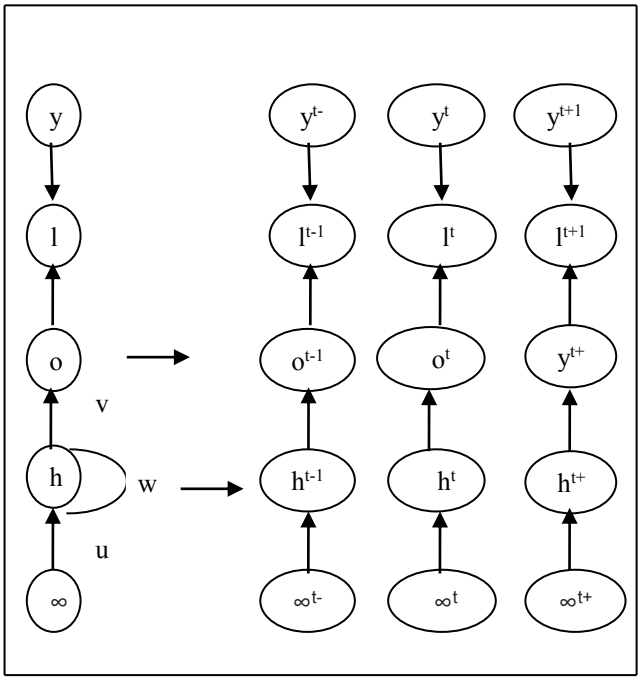


Fig.1.Recurrent Neural Network architecture

B. *Long Short Term Memory*

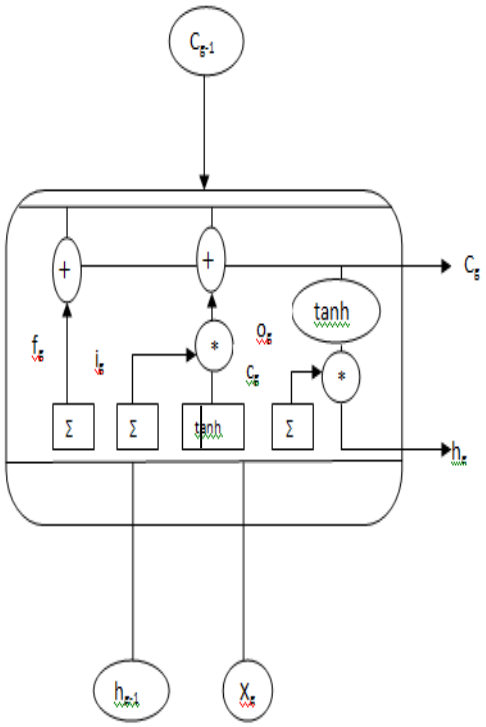
Long Short Term Memory (LSTM) is designed to tackle the long term dependencies it does not cause vanishing gradient problem because LSTM uses the separate memory cell state to carry the information for long time. By using the cell state it carry the information even if the information is not used for long time. LSTM contains three gates forget gate, update gate and reset gate.[8]

The forget gate is to decide what information is need to be thrown away from cell. The update gate decides what information is needed to be stored in cell state. The output gate decides the information that is given as output to the next layer.[13]

The first layer is called as forget layer (f_g) where information from previous layer (h_{g-1})and current input (x_g) with addition of weight (w_f) and bias (b_f) along with sigmoid function (σ) are given in the following formula

$$f_g = \sigma(w_f[h_{g-1},x_g] + b_f) \quad \textbf{(1)}$$

The second layer is input layer (i_g) also called as update layer where input from previous layer and current input is combined and updates the information which is needed. The weight (w_i) and bias(b_i) for update layer is added and functionality of this layer is given as follows[8]



Fi

Fig. 2. Long Short Term Memory Architecture

$$i_g = \sigma(w_i[h_{g-1}, x_g] + b_i) \quad \textbf{(2)}$$

The third layer generated by tanh is called as old cell state (C'_g) where the updated information from input layer is stored. The functionality of new and old cell state is given as follows

$$C'_g = \tanh (w_c . [h_{g-1}, x_g] + b_c \quad \textbf{(3)}$$
$$C_g = f_g * C_{g-1} + i_g * C'_g \quad \textbf{(4)}$$

The final layer is the output layer (O_g) that decides which part of the cell state is given as actual output (h_g) to the next layer [3]. The functionalities of output layer is given as follows

$$O_g = \sigma (W_o [h_{g-1}, x_g] + b_o) \quad \textbf{(5)}$$

Based on the above equation the LSTM model will predict the future sequence by consider the current state and previous state as the input.[13]

IV. ARCHITECTURE OF PROPOSED WORK

The input is a sequence of admissions each admission contain the diagnoses code and intervention code explained about the disease and treatment histories of patients. The codes are embedded using vectors. The GRU architecture diagram for proposed work is illustrated in figure 3. [5]

This approach is to embed admission into vectors. Let D be the set of diagnoses code indexed from 1 to D and I be the set of intervention code indexed from 1 to I. Each admission (t) contains a D: d1,d2……dn, ranges from 1 to D and I : p1,p2,……..pn ranges from 1 to P.

Update Gate

Update gate takes the previous illness history (h (t-1)) and current illness state (x(t)) and previous intervention histories p(t-1).[12]

$$U(t) = \sigma (h(t-1) + x(t) + p(t-1)) \quad \textbf{(6)}$$

Reset gate

Reset gate takes h (t-1) and x (t) as the input and perform sigmoid operation, and it returns R (t). [14]

$$R(t)\ \sigma (h(t-1) + x(t)) \quad \textbf{(7)}$$

Hadamard product is performed for same dimensional vector R (t) and h (t-1), and this result a (t) which determines what to be removed from the previous illness state. And e (t) is added with current input x (t).And now e(t) undergoes tanh activation and return h'(t)

$$h'(t) = \tanh (x(t) + (R(t) \odot h(t-1))) \quad \textbf{(8)}$$

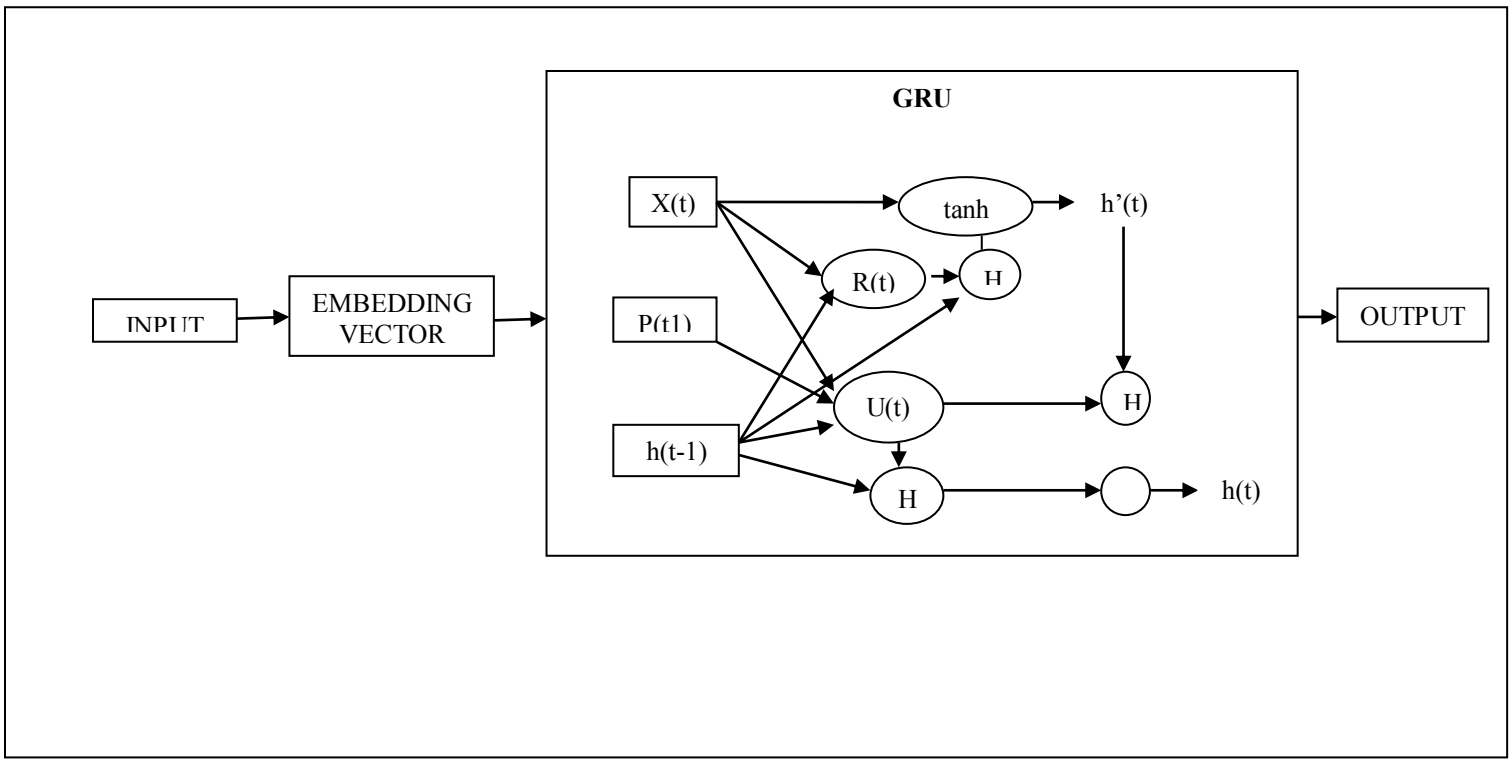


Fig.3.Deep care Gated Recurrent Unit (GRU)

Apply Hadamard function to 1-U (t) and h' (t) and also for U (t) and h (t-1) and sum the results to get the current illness state[14]

$$h(t) = (1- U(t) \odot h'(t) + U(t) \odot h(t-1)) \quad \textbf{(9)}$$

Using this deep care GRU prognosis of diabetes disease have been predict and it also gives better accuracy compared to deep care LSTM model.[5]

V. EXPERIMENTAL SETUP

Anaconda Navigator is the platform for different conda packages it is the Graphical User Interface (GUI) with many machine learning libraries and applications like jupyter notebook, spyder etc. Theano and Tensorflow are python based deep learning libraries whereas, theano is GPU based library and tensorflow is both GPU and CPU based library. It acts as a backend for Keras library. Keras is one of the deep learning library which supports for implementing prepackaged complicated architectures like RNN, LSTM and GRU etc. Tensorflow does not have many prepackaged architecture, it supports to design new architecture but keras support new datasets for known architecture. In this work keras and tensorflow libraries in jupyter notebook application is used. And also keras and tensorflow contains many prepackaged deep learning parameters like activation function, loss function[15].By using these parameters the architecture gives the better results. The parameters are explained as follows

Activation Function

Activation function is used to convert an input signal to the output signal. And this output signal is act as the input signal for the next layer. It uses weight function along with current state input and previous state output and process this parameter. The activation function is broadly classified into two types

- Linear activation function
- Nonlinear activation function

Linear activation function defines that the output will not be inbounded in any specific range. Nonlinear activation function does not have any specific impound or range.

Sigmoid Function: Sigmoid is mainly used when the output is in the form of probability. That is the output is either 0 or 1. The sigmoid curve is in the form of S shape. The sigmoid function can be calculated using the below formula..

$$f(x) = 1 / 1 + \exp(-x) \quad \textbf{(10)}$$

Tanh function: Tanh is similar to sigmoid activation function, it ranges from (-1, 1).compare to sigmoid function tanh can have more gradient strength. But vanishing gradient problem still exist in both tanh and sigmoid function. The tanh activation function is illustrated in equation 11.

$$f(x) = 1\text{-}\exp(-2x) / 1 + \exp(-2x) \quad \textbf{(11)}$$

Relu activation function: Relu is simple and efficient function. And Relu is more better than tanh activation function also it avoids vanishing gradient problem. The relu function is formulated in equation 12.

$$A(x) = \max(0, x) \quad \textbf{(12)}$$

Loss Function

Loss function is used to calculate the inconsistency between predicted value and actual value. The robustness of the model increases if loss function decreases.

Binary cross entropy: Binary cross entropy is a special case of categorical cross entropy. It uses only two classes for classification. The experimental determination of loss functions in using loss function.

Hidden Layers

Hidden layer is present between the input layer and output layer. The output of one hidden layer is the input of next layer. The hidden layer is a set of neurons which is used to process the input by adding some weight function.

Optimization Function

Optimization function is used to train the neural network. Optimization function helps to minimize the error.

VI. IMPLEMTATION RESULTS

C. Dataset Descrription

Diabetes dataset[4] is used and 60% data is for training the data remaining is used for testing the model. The diabetes data for Deep GRU model gives the accuracy 70% and deep care LSTM gives 60% accuracy.

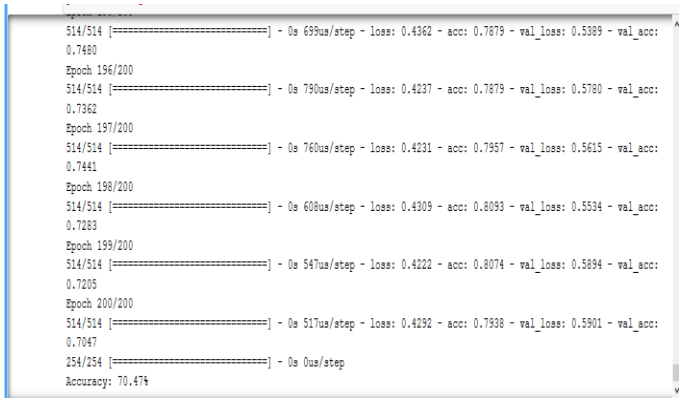


Fig.4.Deep care GRU result

VII. CONCLUSION

Deep learning can be used in many deep learning applications. Inn medical field the disease prognosis is predicted using deep learning. Here prognosis prediction for diabetes disease prediction was studied using Deep care GRU model. And the future progress of work is based on the Gated recurrent unit with its variants.

REFERENCES

[1] Kajaree Das, Rabi Narayan Behera, "A Survey on Machine Learning: Concept, Algorithms and Applications", International Journal of Innovative Research in Computer and Communication Engineering Volume No.02 (2017) pp 2320-9798
[2] R. Sathya Annamma, "Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification", International Journal of Advanced Research in Artificial Intelligence, Volume No 2 (2013) pp 300-306
[3] Junyoung Chung Caglar Gulcehre KyungHyun Cho Universite de Montreal ´ Yoshua Bengio, Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling, Journal of Cornell university Volume No.02 (2014) pp 200-206
[4] Benjamin Shickel, Patrick J. Tighe, "Deep EHR: A Survey of Recent Advances in Deep Learning Techniques for Electronic Health Record (EHR) Analysis", Journal of Cornell university (2018)
[5] Trang Pham, Truyen Tran, Dinh Phung, Svetha Venkatesh, "Predicting healthcare trajectories from medical records: A deep learning approach", Journal of Biomedical Informatics Volume No. 69 (2017) pp 218–229
[6] Cao Xiao1, Tengfei Ma2, Adji B. Dieng3, David M. Blei3, Fei Wang,"Readmission prediction via deep contextual embedding of clinical concepts", Research article Plos one 2017
[7] Ioannis Kavakiotis Olga Tsave , Athanasios Salifoglou c, Nicos Maglaveras Ioannis Vlahavas , Ioanna Chouvarda, "Machine Learning and Data Mining Methods in Diabetes Research", Computational and Structural Biotechnology Journal Volume no 15 (2017) pp 104–116
[8] Zachary C. Lipton ," Learning To Diagnose With Lstm Recurrent neural Networks", ICLR(2016)
[9] Joseph Futoma , Jonathan Morris , Joseph Lucas, "A comparison of models for predicting early hospital readmissions", Journal of Biomedical Informatics Volume No 56 (2015) pp 229–238
[10] Veena Vijayan V. Anjali C., "Prediction and Diagnosis of Diabetes Mellitus –A Machine Learning Approach",IEEE Recent Advances in Intelligent Computational Systems (RAICS) ,Volume No 43 Page no 10-12
[11] Truyen Tran , Tu Dinh Nguyen , Dinh Phung , Svetha Venkatesh , "Learning vector representation of medical objects via EMR-driven nonnegative restricted Boltzmann machines (eNRBM)", Journal of Biomedical Informatics Volume no 54 (2015) Page no 96–105
[12] Junyoung Chung Caglar Gulcehre KyungHyun Cho Universite de Montreal ´ Yoshua Bengio, Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling, Journal of cornell university Volume No.02 (2014) pp 200-206

WEB REFERENCES

[13] http://colah.github.io/posts/2015-08-Understanding-LSTMs/
[14] https://towardsdatascience.com/understanding-gru-networks-2ef37df6c9be
[15] NMT-Keras Documentation - https://keras.io/