Effective Deep Learning Model to Predict Student Grade Point Averages

Conference Paper · December 2017

DOI: 10.1109/ICCIC.2017.8524317

Effective Deep Learning Model to Predict Student Grade Point Averages

Akhilesh P Patil, Karthik Ganesan, Anita Kanavalli
Department of Computer Science and Engineering
Ramaiah Institute of Technology
Bengaluru, India
Apatil1997@gmail.com, Karthikganesan@gmail.com, Anithak@msrit.edu

Abstract— **The main objective of this paper is to provide an overview of the deep learning techniques that can be used to predict a student's performance as compared to other traditionally used machine learning techniques. In our research, we developed feed forward neural networks and recurrent neural networks for developing a model to effectively predict the student GPA. The recurrent neural networks gave greater accuracy as compared to feed forward neural networks, as they have memory and take into consideration the consistency of the student performance. The main contribution of the paper is that we have compared various recurrent neural architectures such as single hidden layer long short-term memory network, long short-term memory network with multiple hidden layers, and Bi-directional long short-term memory network with multiple hidden layers. We compared these techniques with root mean square error as the parameter of comparison and found Bi-directional long short-term memory network to have the least error of 8.2%. A comparison of results of the proposed technique versus other deep learning models and machine learning techniques has been provided in the section VIII and the visualization of the results has been provided in section IX. The novelty of the method proposed is that it has memory to differentiate tuples with different order of scores and learn to assign the weights of relationship between nodes by scanning the sequence in both directions compared to decision tree, SVM, feed forward neural network based algorithms which have been earlier used to solve this problem of predicting student score.**

Keywords— Artificial Neural Networks, Recurrent Neural Networks, Long Short Tertm Memory Networks, Bidirectional Long Short Term Memory Networks, Root Mean Square Error.

I. INTRODUCTION

Predicting the outcome of any task which has been undertaken by an individual is a very interesting topic which has caught the attention of people today. The technique of analysing the past events to predict the outcome of the future is known as Predictive Analysis. Predictive Analysis incorporates many techniques such as data mining, machine learning, statistical modelling and artificial intelligence to analyse current data to make predictions about the future. Using these techniques, we can predict quite accurately the outcome of the student's grade. As we all know, most of the colleges and schools in India today follow the CIE (Continuous Internal Evaluation) system to assess and grade their students. In this type of grading system, students receive a final grade based on how they perform in their respective internal examination as well as their final semester end examination. After clearing all the internal examinations, students can predict the outcome of their final grade.

The huge data that is available to us may contain certain trends and patterns. These trends play a key role in identifying the prospects of the future. The task at hand is to identify the relationships between these prospects and assign a score to each of them. This process of predicting the score can be accurately done by computers by using the modern techniques of Machine Learning. There are various methods to approach this problem to predict the final grade of students. Using deep learning techniques, one can come up with a model to solve this problem. This paper gives a brief description of the use of LSTM (Long Short-Term Memory) Networks which is a type of Recurrent Neural Network to come up with a model to solve the problem.

II. OBTAINING THE DATASET AND DATA PREPROCESSING

The dataset which we train our machine learning model is basically a register of marks containing the marks of 2000 students in five subjects and the expected grade point averages. These marks are randomly generated according to a certain logic and the grades are also given according to a certain grading methodology. The marks are present in the form of a csv file, which can be read using the pandas library in python. The pandas library is used to make a data frame out of the csv file, making it easier to manipulate. For pre-processing of the data, the Minmax Scalar method present in the pre-processing class of the sklearn library was suitable. The training set and the test set was split on an 80:20 basis. The input variables and the output target variables are the marks in various subjects and semester grade point averages respectively. The snapshot of the simulated dataset is as depicted in the below table:

TABLE I. SNIPPET OF DATASET

Sub1	Sub2	Sub3	Sub4	Sub5	CGPA	SGPA
28	26	27	32	33	6.4	6.43
33	36	39	42	38	7.4	7.06
38	38	43	37	44	7.8	7.17
34	33	38	40	44	7.4	7.14

978-1-5090-6621-6/17/$31.00 ©2017 IEEE

42	43	47	42	43	9.2	9.47
37	42	36	46	44	8.72	8.7
44	45	46	47	48	9.7	9.86

III. GRADING METHODOLOGY

A random dataset was generated taking into consideration a particular pattern of grading methodology as follows: There are two types of examinations which a student has to take up. The first one is an internal examination which comprises of 50 marks and a semester end examination which comprises of 100 marks. The internal examination is further split into 30 and 20 where, the 30 marks is a written test and 20 marks is based for some mini projects, assignments etc. The semester end examination which is for 100 marks is scaled down to 50 and added to the 50 marks of internal assessment to finalize the grade of a student in each subject. The grading scale for a batch of students is as follows:

TABLE II. GRADING METHODOLOGY

Range of Marks	**Grade**	**Level**	**Grade Point**
Marks >= 90%	S	Outstanding	10
75% <= Marks < 90%	A	Excellent	9
60% <= Marks <75%	B	Very Good	8
50% <= Marks <60%	C	Good	7
45% <= Marks <50%	D	Average	5
40% <=Marks <45%	E	Poor	4
Marks < 40%	F	Fail	0

IV. LITERATURE SURVEY

There were a few related research articles which come close to our topic of discussion. As cited in [1], the research compares the accuracies of various machine learning algorithms using true positive rates, false positive rates and thereby computing the precision and recall. As discussed in our research we aim at comparing the accuracies of a few machine learning algorithms and deep learning algorithms using the root mean square error. The root mean square error gives a much clear and direct way of finding the accuracy.

In another paper cited in [2], feed forward neural networks, Bayesian classifier and Support Vector Machines were used for comparison. Through our research, we have shown that use of recurrent neural networks and bidirectional networks have an upper hand over feed forward networks. We have also compared these networks with machine learning algorithms cited in [2] and have shown use of Long Short-Term Networks are much accurate.

Several other researches cited in [3] and [4] use classification techniques such as decision trees and other regression techniques to predict student grade point averages. Through the technique used in this paper, the results show that use of bidirectional networks is far more advantageous than the techniques cited above.

V. RECURRENT NEURAL NETWORKS.

The idea behind Recurrent Neural Networks is to make use of sequential data. Recurrent Neural Networks are called so because for each of the elements in a sequence (the sequence of subject marks in our case) the output is calculated based on the previous combination. Recurrent Neural Networks can be considered as having the memory of each of the subject marks calculated so far. Theoretically, RNN's make use of long sequences of data, but in practice they are limited to looking back only a few steps. This way, we can take into account the consistency of marks scored by a student. In our case we use a sequential array of marks as input data for the neural network. As the neural network computes the same task for every element in the sequence, the network keeps a track of the all the records of the student to finally come up with the best predicted score. In theory, the neural net keeps a memory of the track record of the student.

In our case we use a sequential array of marks as input data for the neural network. As the neural network computes the same task for every element in the sequence, the network keeps a track of the all the records of the student to finally come up with the best predicted score. In theory, the neural net keeps a memory of the record of accomplishment of the student.

A. Advantages of Recurrent Neural Networks over Feedforward Networks.

One of the major limitations in a feed forward network is that it has no memory. Each prediction made by such a network is independent of the previous calculations made. But for our task, which consists of a set of grades as inputs, the predicted outcome may depend on the previous grades of the student. Hence a recurrent neural network may be of more significance here.

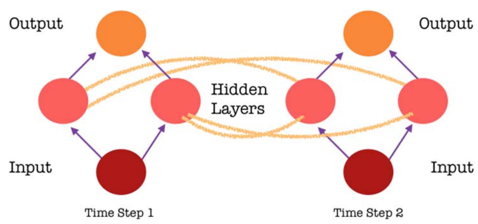

Fig 1. Recurrent Neural Networks.

A recurrent neural network includes connections between neurons in hidden layers that can feed back upon themselves. These hidden layer links one-time step to another one. This forms a type of short term memory. This brings into discussion a type of Recurrent Neural Networks called Long Short-Term Memory Networks.

B. How can Recurrent Neural Networks be used to detect abnormalities.

As we know, a recurrent neural network has a looping methodology between its hidden layers which keeps track of the previous results before coming up with a final predicted score. This way we can keep track of the consistency maintained by a student in a particular semester as well as his consistence throughout his semester history. On comparing the test values of the dataset corresponding to a student's grade point averages and the predicted values we can make out abnormalities like discrepancies in marks entry, underperformance of a student and some critical cases. A deviation of grade points over a certain threshold will help in identifying the abnormalities.

VI. LONG SHORT-TERM MEMORY NETWORKS

Humans do not start to think from scratch every time they read some new article. As a person reads an article, he or she reads every word based on the understandings of the previous words. Our thoughts can retain the things of the past. A simple neural network will not have the ability do so. Recurrent neural networks can overcome this shortcoming, due to the presence of a looping methodology in the hidden layers. Long Short-Term Memory networks are a special kind of Recurrent Neural Networks, which have the capability of learning from previously computed output results. In our problem where we predict the grades of the students, the whole idea here is that the grades in our data-frame are a sequence of marks which we may consider as a vector. The RNN that obtains the vectors as input and considers the order of vectors to generate predictions. From the embedding layer, the new representations will be passed to LSTM cells. These will add recurrent connections to the network, so we can include information about the sequence of marks collected. Finally, the cells of the LSTM will go to sigmoid output layer. We use a sigmoid because we are trying to predict the final grade. Unlike artificial neural networks Long Short-Term Memory Networks do not have a single layer of neural network. Instead, they have four interacting and repeating modules of chain like structures. The basic structure of the LSTM architecture is as shown in the below figure.

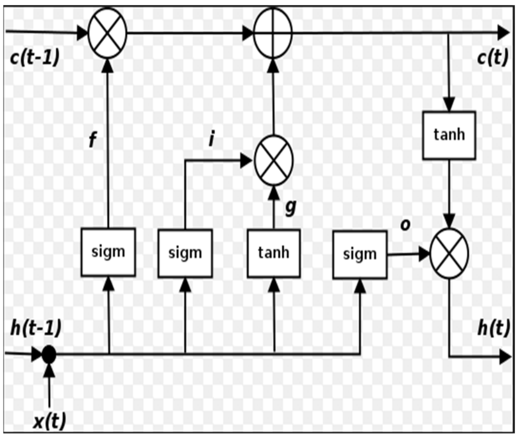

Fig 2. Long Short-Term Memory Network.

In the first step, the LSTM model decides by a sigmoid layer, looks at a previously computed hidden layer- h_{t-1} and the present output- x_t and outputs a number between 0 and 1 for each number in state C_{t-1}.

$$f_t = \sigma(W_f[h_{t-1}, x_t + b_f]) \tag{1}$$

Secondly, we need to determine as to what is going to be stored in the successive cell. Firstly, the sigmoid layer decides as to which values to update. Finally, the tanh layer vectorizes the values to form the C_{t1} values. The next step involves the combination of the two layers.

$$i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i) \tag{2}$$

$$C_{t1} = \tanh(W_c.[h_{t-1}, x_t] + b_c) \tag{3}$$

In the next step, we update the old cell state C_{t-1} to new state C_{t1}. In the case of our model this is where we drop the information of the old state and add new information, as decided earlier. Here, C_{t-1} is the previous old cell state.

$$C_{t1} = f_t * C_{t-1} + i_t * C_t \tag{4}$$

Finally, we come up with the output, that is the final information of a sentence. For the model, it might want to output information relevant the outcome of the result.

$$o_t = \sigma(W_o.[h_{t-1}, x_t] + b_o) \tag{5}$$

$$h_t = o_t * tanh \tag{6}$$

VII. ROOT MEAN SQUARE ERROR

We can check the accuracy of the recurrent neural network by using a statistical method called root mean squared error. In this methodology the exact difference between the observed frequency and the predicted frequency of values can be determined. The individual differences can be considered as residuals, and the RMSE serves to aggregate them into a single measure of predictive power. The RMSE of a model prediction with respect to the estimated variable X model is defined as the square root of the mean squared error. For our Long Short-Term Memory Network model, the residual values were founded for the test values and the predicted grade values. The following equation is used to find the Root mean square error.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(X_{obs,i} - X_{model,i})^2}{n}} \tag{7}$$

where X_{obs} is observed values and X_{model} is modelled values at time/place i.

VIII. IMPROVING ACCURACY USING BIDIRECTIONAL LSTMs

Bidirectional Long Short-Term Memory Networks (LSTMs) can extract the most out of an input sequence by running over the input sequence both in the forward direction as well as in the backward direction. The architecture is developed by creating another copy of the first recurrent layer in the neural network so that there are two layers side-by-side. The next step is to provide the input sequence as it is to the first layer and feeding an inverted copy to the second duplicated layer. This approach is effectively used along-side the Long Short-Term Memory Networks.

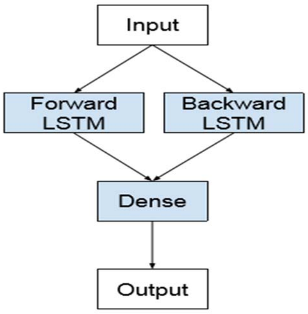

Fig 3. Bidirectional Long Short-Term Memory Networks.

IX. CONCLUSION AND RESULTS

Through the predictive model which uses a type of Recurrent Neural Network called the Long Short-Term Memory Network it was possible to predict the final grade of a student. Considering the use of Recurrent Neural Networks, the model also predicts based on the consistency of the student's performance in his previous semesters. This way we can also ascertain if there are any discrepancies in the entry of the final marks. The Long Short-Term Memory Networks were trained using four hidden layers, with 11 rows of data for 2000 students. This makes the model much more accurate.

The root mean square error methodology is used to show the difference between the actual grade point average of the student and the predicted grade point average of the student. Initially, the data was fed to a feed forward network which is the artificial neural network using back propagation. The accuracy was found to be 68 percent.

The root mean squared error for the single layered Long Short-Term Memory Neural Network was found to be around 24 percent. From this, we can tell that our model is around 76 percent accurate.

Using multiple layers-five to be precise, the root mean squared error was tuned up-to 15 percent. This indicates that the model was found to be 85 percent accurate.

Incorporating bidirectionality in the Long Short-Term Memory Networks the root mean squared error was decreased to 7.4 percent. Hence, an accuracy of 92.6 percent was obtained. Since the bidirectional Long Short-Term Memory Networks were found to be more accurate, we can choose this model to come up with the statistics of student data.

We could obtain the number of students whose marks were predicted with 5 percent error 10 percent error and 20 percent error. Out of the 2000 students, 1357 student marks were predicted with less than or equal to 5 percent error, 238 student marks were predicted with less than or equal to 10 percent error and 5 student marks were predicted with less than or equal 20 percent error. We also compared the Root Mean Square Error of Machine Learning techniques like Decision Tree, Support Vector Machine, Random Forest and Linear Regression. Through our comparisons we were able to show that the Bidirectional Long Short-term Memory networks used in our research showed the best result.

We can compare the results by observing the values of the part of the dataset used as test data and the predicted final scores of the students in respective. The test and the predicted values for both the above-mentioned cases are as depicted in the below figure:

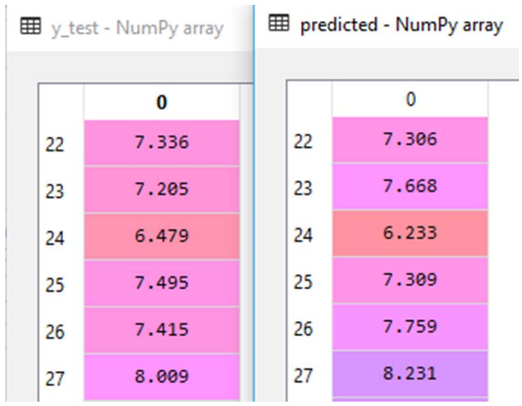

Fig 4. Comparison of Results

The research idea and implementation proposed through this paper may adhere with the grading system of one college. However, the implementation can be altered based on the needs of a grading system. This can be seen in few other research papers which have depicted a similar idea. There were a few related research articles which come close to our topic of discussion, a few of which are: Student academic performance monitoring and evaluation using data mining techniques, Comparative study of artificial neural networks and statistical models for predicting student grade point averages, Student Performance Prediction Model using Machine Learning Approach: The Case of Wolkite University and a few more mentioned in the reference section. However, the techniques used in the above mentioned were individually different from the techniques used in our model. Through our model, we are not only able to come up with the predicted score for students, but also reveal if there are any discrepancies or human error in entry of marks.

A table of comparison is shown as following.

TABLE III. COMPARISON OF DEEP LEARNING ALGORITHMS

Neural Network Architecture	Error	Accuracy
Feed Forward Network	32.4%	67.6%
Single Layer Long Short-term Memory Networks	24.3%	75.7%
Multi-layered Long Short-Term Memory Networks	14.8%	85.2%
Bi-directional Long Short-Term Memory Networks	7.4%	92.6%

TABLE IV. COMPARISON OF MACHINE LEARNING ALGORITHMS

Algorithm	Error	Accuracy
Decision Tree	20.91%	79.09%
Support Vector Machine	31.39%	68.61
Random Forest	15.33%	84.67%
Linear Regression	31.66%	68.34%

X. VISUALISATION

A visualisation was also obtained for comparing the predicted and the actual values. The X axis in this visualisation depicts the student ID and the Y axis represents the grades. The visualisations were done for three models used for comparative analysis. The blue shade indicates the predicated values and the orange shade indicated the actual values. The visualisations were carried out for the more accurate main deep learning algorithms that is Bi-directional Long Short-Term Networks and Multi-Layered Long Short-Term Memory Networks.

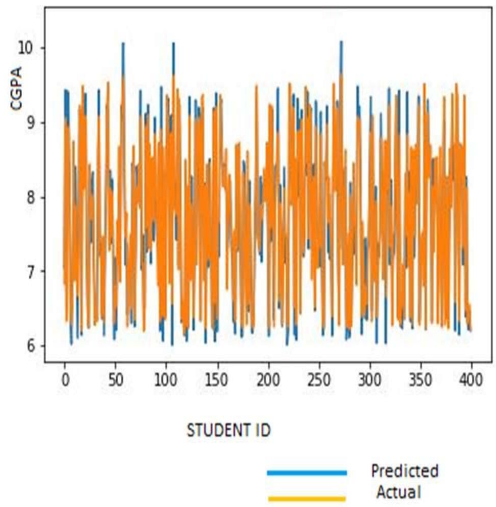

Fig 5. Visualisation of Multi-Layered Long Short-Term Networks.

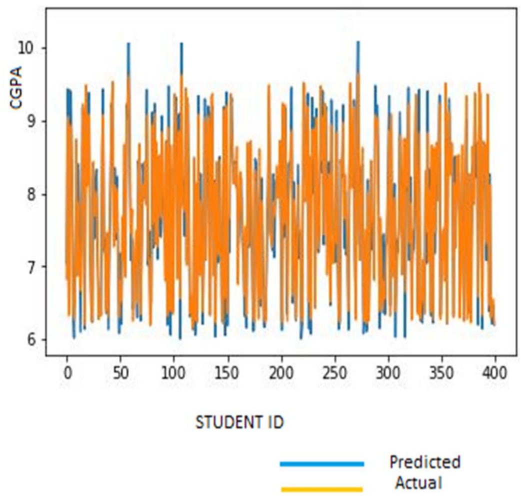

Fig 6. Visualisation of Artificial Neural Network

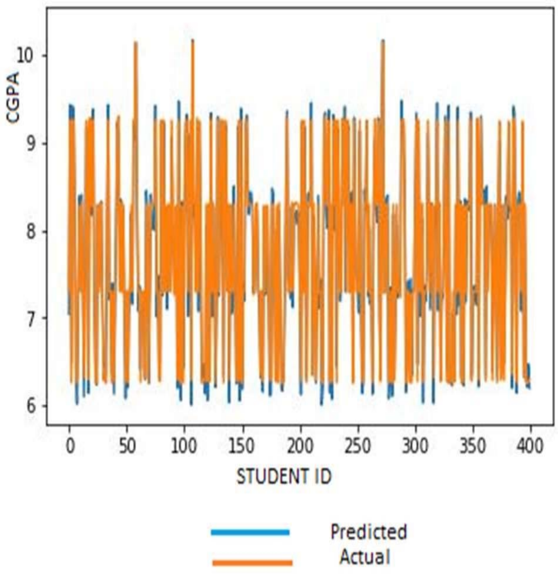

Fig 7. Visualisation of Bi-directional Long Short-Term Networks.

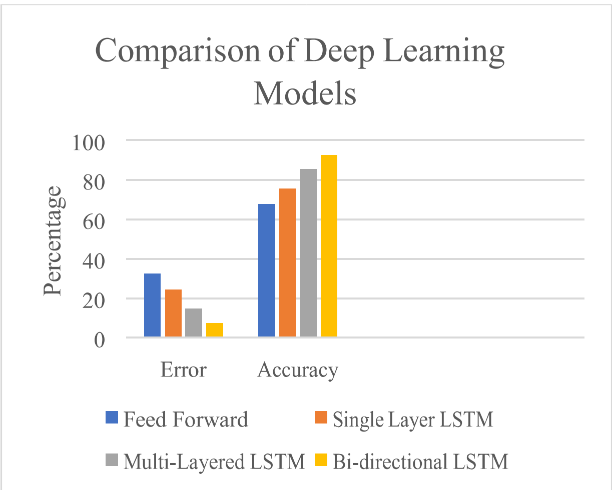

Fig 8. Graph of Percentage Accuracy for Deep Learning Models

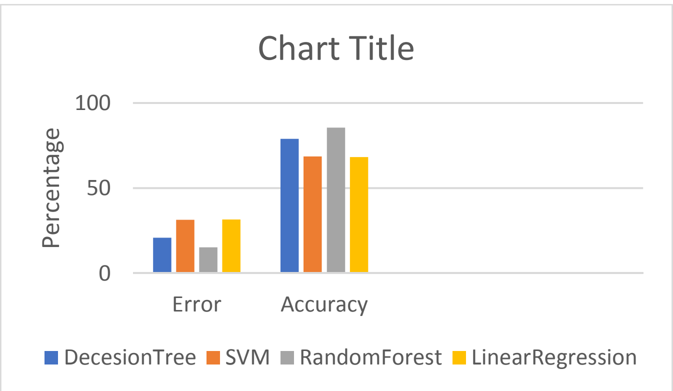

Fig 9. Graph of Percentage Accuracy for Machine Learning models.

REFERENCES

[1] Emaan Abdul Majeed, and Khurun Nazir Junejo, "Grade Prediction Using Supervised Machine Learning Techniques," e-Proceedings of the 4th Global Summit on Education 2016.

[2] Ermiyas Birihanu Belachew, and Feidu Akmel Godena, "Student Performance Prediction Model using Machine Learning Approach: The case of Wolkite University", in International Journal if Advanced Research in Computer Science and Software Engineering: Volume 7, Issue 2, Febuary 2017.

[3] Mashael A. Al-Barrak, and Muna Al-Razgan, "Predicting Students Final GPA using Decesion Trees: A Case Study", in International Journal of Information and Education Technology, Vol 6, No 7, July 2016.

[4] Janusz Szczypula, "Comparitive Study of artificial neural network and statistical models for oredicting student grade point averages", in International Journal of Forecasting, Volume 10, Issue 1, June 1994.

[5] Colah's Blog, "Understanding LSTM Networks", August 27, 2015.

[6] Zafar Iqbal*,Junaid Qadir**, Adnan Noor Mian* and Faisal Kamiran, Machine Learning Based Student Grade Prediction: A Case Study.

[7] Havan Agrawal, Harshil Mavani, Student Performance Prediction using Machine Learning, International Journal of Engineering Research & Technology (IJERT) Vol. 4 Issue 03, March-2015.

[8] Jie Xu, Member, IEEE, Kyeong Ho Moon, Student Member, IEEE, and Mihaela van der Schaar, Fellow, IEEE, A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs