International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 06 | June 2021 www.irjet.net p-ISSN: 2395-0072

Student Course Prediction using Deep Learning techniques

Divit Raina[1], K P Supreethi[2]

[1]*Student, Department of Computer Science and Engineering, JNTUH College of Engineering, Hyderabad, Telangana, India-500085*
[2]*Professor, Department of Computer Science and Engineering, JNTUH College of Engineering, Hyderabad, Telangana, India-500085*

Abstract-*As students prepare for establishing themselves in the industries, it is crucial that they get familiar with their preferred domains beforehand ,along with their academics. Nowadays, with the availability of plenty of online courses ,a student can gain knowledge in any domain of choice. The student course predictor helps the student by suggesting courses to students based on their profiles.. Through the comparison of different approaches of neural networks ,ie,Dense neural networks and Convolutional neural networks(with DNN),models are built ,using the functionalities of libraries such as Tensroflow, for the task of suggesting courses to the students based on their current profile. The model focuses on students of Computer Science branch.*

Key words- **Dense Neural Networks, DecisionTrees, Student course predictor, Tensorflow, One hot encoding, sklearn, Convolutional Neural Networks**

INTRODUCTION

In today's world, a student should not only be good in academics but should also be familiar with different techniques and technologies used in the industries where they wish to work in the near future. This knowledge can be gained from doing some internships and courses, so it is really important for students to do some courses as well along with studying their regular subjects. With the vast availability of courses available to students, the students have a lot of courses to choose from. With the help of this model, the students can be suggested courses in different fields based on their current profile.

RELATED WORK

In many similar projects such as predicting a student's academic performance based on their profiles and predicting the industry where a student should work in ,based on their profile and interests ,many similar approaches using machine learning techniques are used apart from neural networks ,which is used here. They include: Support vector machines(SVM),XG Boost, Naïve bayes algorithm, K-Nearnest neighbors algorithm, Decision trees

Support Vector Machines(SVM)

Support Vector Machine (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems

This approach has been used in similar problems such as predicting the academic performance of a student ,measuring the performance in the form of labels which is a classification problem.

XG Boost algorithm

XGBoost denotes eXtreme Gradient Boosting. XGBoost is implementation of gradient boosting algorithms It mainly focuses on model performance and computational time. It greatly reduces the time and greatly lifts the performance of the model .It has been used for predicting students career area based on the current profile of the student ,having a better accuracy as compared to other approaches such as SVM.

Naïve bayes algorithm

The Naïve bayes algorithm has also been used in tasks like Student performance prediction based on the profiles of various students. This approach has computational simplicity as the model is easy to be trained and tested along with having a decent accuracy on predicting the class labels.

K-Nearest Neighbor algorithm

KNN algorithm used for classification and regression prediction. It is commonly used for its ease of interpretation and low calculation time In a similar task of predicting the performance of students, this approach of using K-Nearest Neighbor algorithm has been used. It is very easy to implement but it becomes very slow for large datasets.

Decision Trees

In a task of predicting the industry roles of students based on their current profile and interests ,decision trees approach was used .Decision tree builds classification or regression models in the form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes.

PROPOSED METHOD

For the task of predicting or suggesting students what courses they should do,there are two approaches of Neural netowrks used here: Dense Neural networks and Convolutional neural networks(with a few dense layers) .

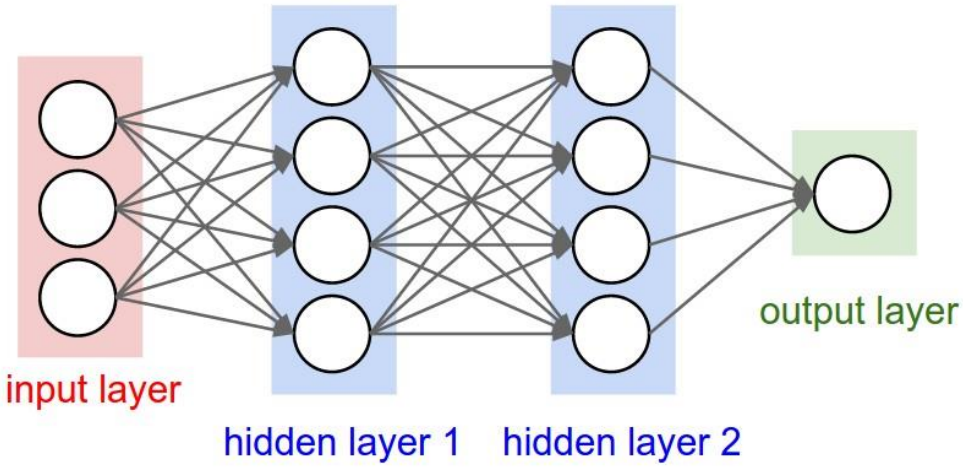


Figure 1: A neural network (Source: https://in.pinterest.com/pin/516928863484589618/)

Neural Networks

In electronics engineering and related fields, artificial neural networks (ANNs) are mathematical or computational models that are inspired by a human's central nervous system (in particular the brain) which is capable of machine learning as well as pattern recognition. So neural networks, with their stronger ability to derive meaning from complicated or imprecise data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques.

- **Dense neural networks:** In Dense neural networks, there are a number of neurons in each layer, receiving input from the neurons in the preceding layer. Since, the input is received from all the neurons of the preceding layer to the neurons in the next layer, the neural networks are "dense".

- **Convolutional neural networks:** The CNN is a neural network with a special structure. CNN the first layer, which consists of a number of feature maps, is called a convolution layer. Each neuron in the convolution layer receives input from a local receptive field representing features of a limited frequency range. Neurons that belong to the same feature map share the same weights (also called filters or kernels) but receive different inputs shifted in frequency. As a result, the convolution layer does a convolution of the kernels with the lower layer activations

Dataset Description

The dataset used for the task has a total of 18 columns with the first 17 columns used as features to predict on the 18th column. It has a total of 20000 records of different students.

Percentag	Percentag	Percentag	Percentag	Percentag	Percentag	Percentag	Percentag	Percentag	Hours stu	Logical qu	Number o	Coding sk	Public spe	Can study	Capable o	Wish to d	Suggested Course
69	63	78	87	94	94	87	84	61	9	4	0	4	8	yes	yes	yes	shell programming
78	62	73	60	71	70	73	84	91	12	7	1	2	3	yes	no	yes	machine learning
71	86	91	87	61	81	72	72	94	11	1	4	1	3	yes	no	yes	app development
76	87	60	84	89	73	62	88	69	7	1	1	2	5	no	yes	no	python
92	62	90	67	71	89	73	71	73	4	5	4	6	3	no	no	no	app development
88	86	62	79	93	84	69	71	82	11	5	3	8	1	yes	yes	no	python
93	77	69	79	90	93	73	63	77	6	3	2	3	3	no	no	yes	r programming
84	72	88	62	66	63	78	94	60	12	2	1	6	6	yes	no	no	r programming
73	66	66	81	81	69	61	87	90	10	5	2	4	8	no	yes	no	information security
62	76	85	91	82	69	63	63	81	10	9	0	5	4	no	no	yes	app development
73	80	70	83	61	82	75	94	89	8	6	2	3	9	yes	no	no	hadoop
63	76	81	90	70	72	86	89	85	8	3	6	9	7	no	yes	no	machine learning
63	64	61	67	61	70	65	64	79	8	2	2	8	6	yes	no	yes	distro making
68	93	77	60	77	81	67	81	91	12	5	3	6	5	no	no	yes	python
90	83	66	71	65	67	86	63	62	9	6	2	3	8	yes	no	yes	machine learning
94	71	63	74	61	69	65	62	68	4	7	6	5	8	yes	yes	yes	information security
60	92	94	90	62	94	62	73	60	11	7	5	5	4	no	no	yes	app development
92	86	61	63	64	65	92	65	60	4	9	6	3	4	no	yes	no	hadoop
82	91	68	86	78	61	91	82	82	12	8	0	7	1	no	no	no	machine learning
90	73	90	67	63	88	67	60	69	12	6	0	2	8	no	yes	no	hadoop
67	61	66	62	93	91	88	88	70	4	5	0	2	5	yes	yes	yes	app development
65	61	73	90	89	74	82	60	84	5	8	3	2	8	yes	yes	yes	hadoop
84	62	70	70	67	65	78	61	65	4	9	6	4	5	no	no	no	r programming
74	89	66	75	86	61	63	80	69	9	9	5	7	6	no	yes	yes	shell programming

Table 1: Dataset used

The first nine column consists of the percentages of students in different core subjects of Computer science (ranging from 0 to 100).The next column denotes the maximum number of hours that the particular student is willing to study or can study.Next column denotes the logical quotient rating of the student(from 0 to 10),followed by the column denoting the number of hackathons attended by the students. The next two columns denote the coding skills and public speaking skills of the students (from 0 to 10) .The next three columns consists of categorical values (yes or no).These columns indicate whether a student is willing to work/study long time before computer screens ,whether the students are capable of self learning and whether they really wish to do some courses apart from their academics to gain extra-knowledge. The final column consists of the courses, which are suggested to the students by the model.This column includes nine district classes (shell programming, machine learning, app development, python,r programming, information security ,Hadoop, distro making and full stack)

IMPLEMENTATION

Firsly, the features and labels of the dataset mentioned above are to be converted to a suitable format before it can be used to train the neural network.

Using Numpy slicing, the features and the labels(last columns) are extracted.

Now, to convert the data into a suitable format for feeding into the neural network:

Features

The first 14 columns (or features) ,which contains percentages of students in different subjects ,Hours a student can work,logical quotient rating ,number of hackathons attended by the students ,coding skill and public speaking ratings are normalized(using the MixMaxScaler of the sklearn library) so that they lie with the range of [0,1].

The next three columns(Can study/Work long time before computer screens, Capable of Self-Learning, Wish to do more extra courses(apart from academics)) contain only the values 'yes'or 'no'.So,in order to feed it to the neural network,these non numerical labels are converted to numerical labels using the LabelEncoder() of sklearn library.

Percentage in Operating Systems	Percentage in Algorithms	Percentage in Programming Concepts	Percentage in Software Engineering	Percentage in Computer Networks	Percentage in Electronics Subjects	Percentage in Computer Architecture	Percentage in Mathematics	Percentage in English language skills	Hours studying per day(Max)	Logical quotient rating	Number of hackathons attended	Coding skills rating	Public speaking points	Can study/Work long time before computer screens	Capable of Self-Learning	Wish to do more extra courses(apart from academics)
0.264706	0.0882353	0.529412	0.794118	1	1	0.794118	0.705882	0.0294118	0.625	0.375	0	0.375	0.875	1	1	1
0.529412	0.0588235	0.382353	0	0.323529	0.294118	0.382353	0.705882	0.911765	1	0.75	0.166667	0.125	0.25	1	0	1
0.323529	0.764706	0.911765	0.794118	0.0294118	0.617647	0.352941	0.352941	1	0.875	0	0.666667	0	0.25	1	0	1
0.470588	0.794118	0	0.705882	0.852941	0.382353	0.0588235	0.823529	0.264706	0.375	0	0.166667	0.125	0.5	0	1	0
0.941176	0.0588235	0.882353	0.205882	0.323529	0.852941	0.382353	0.323529	0.382353	0	0.5	0.666667	0.625	0.25	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
0.676471	0.205882	0.0588235	0.0882353	0.617647	0.411765	0.882353	0.705882	0.264706	0	0.125	1	0	0.875	1	0	0
0.588235	0.264706	0.676471	0.794118	0.647059	0.176471	0.176471	0.941176	0.205882	0.375	0.875	0	1	0.875	1	0	0
0.676471	0.294118	0.588235	0.794118	0.117647	0.735294	0.264706	1	0.823529	0.375	0.25	1	0.125	0.25	1	1	1
0.235294	0.794118	0.911765	0.823529	0.176471	0.411765	0.0294118	0.794118	0.0294118	0.125	0	0.666667	1	0.5	0	0	0
0.382353	0.5	0.411765	0.705882	0.294118	0.147059	0.941176	0.382353	0.882353	0.375	0.25	0.166667	0.75	0.625	1	1	1

Table 2: Result of preprocessing on features

Labels

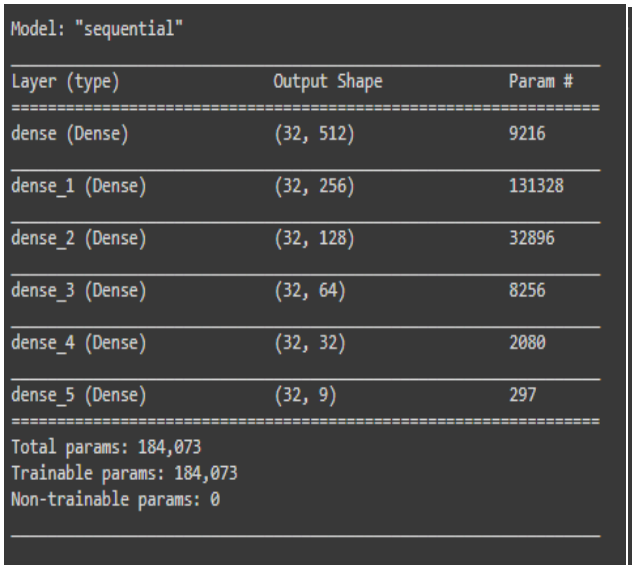
The labels are the values that are to be predicted by the model(the suitable course for the student).So,they are also 'OneHotencoded' using the get_dummies() of the pandas library.

app development	distro making	full stack	hadoop	information security	machine learning	python	r programming	shell programming
0	0	0	0	0	1	0	0	0
0	0	0	0	1	0	0	0	0
0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0	0
0	0	1	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...
0	0	0	1	0	0	0	0	0
0	0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	1	0	0

Table 3: Result of preproccesing on labels

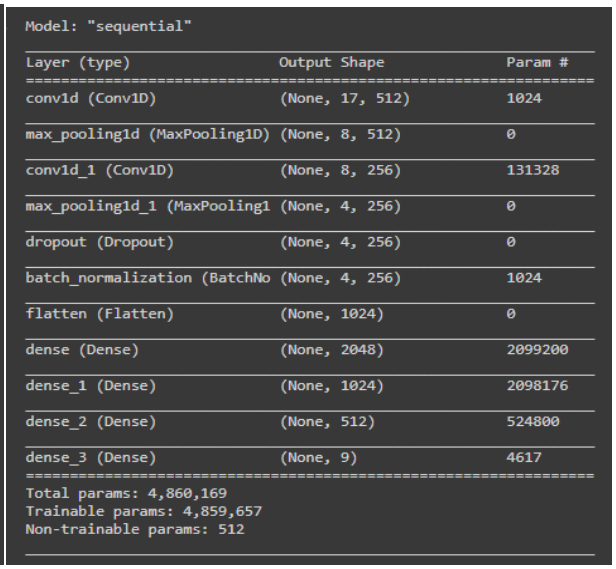
The data is split into training(80%) and testing(20%) using the train_test_split function.

Now,there are two models created for the task of prediction of suitable courses for students.The pre-processed data is fed to the following two models:



```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (32, 512)                 9216
_________________________________________________________________
dense_1 (Dense)              (32, 256)                 131328
_________________________________________________________________
dense_2 (Dense)              (32, 128)                 32896
_________________________________________________________________
dense_3 (Dense)              (32, 64)                  8256
_________________________________________________________________
dense_4 (Dense)              (32, 32)                  2080
_________________________________________________________________
dense_5 (Dense)              (32, 9)                   297
=================================================================
Total params: 184,073
Trainable params: 184,073
Non-trainable params: 0
_________________________________________________________________
```

Layers of Dense neural network



```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv1d (Conv1D)              (None, 17, 512)           1024
_________________________________________________________________
max_pooling1d (MaxPooling1D) (None, 8, 512)            0
_________________________________________________________________
conv1d_1 (Conv1D)            (None, 8, 256)            131328
_________________________________________________________________
max_pooling1d_1 (MaxPooling1 (None, 4, 256)            0
_________________________________________________________________
dropout (Dropout)            (None, 4, 256)            0
_________________________________________________________________
batch_normalization (BatchNo (None, 4, 256)            1024
_________________________________________________________________
flatten (Flatten)            (None, 1024)              0
_________________________________________________________________
dense (Dense)                (None, 2048)              2099200
_________________________________________________________________
dense_1 (Dense)              (None, 1024)              2098176
_________________________________________________________________
dense_2 (Dense)              (None, 512)               524800
_________________________________________________________________
dense_3 (Dense)              (None, 9)                 4617
=================================================================
Total params: 4,860,169
Trainable params: 4,859,657
Non-trainable params: 512
_________________________________________________________________
```

Layers of Convolutional neural network

The Stochastic gradient descent optimizer(with learning rate of 0.01) is used for both the models,along with Categorical crossentropy as loss(as there are 9 distinct classes) ,accuracy and mean squared error are measured as metrics .

RESULT AND DISCUSSION

After training the Convolutional neural network(with dense layers) and the Dense neural network for 300 epochs(using tensorflow library) ,the results for both the models were obtained as follows:

Neural Network	Accuracy	Loss	Validation accuracy	Validation loss	Mean sq. error (MSE) (test data)
Dense neural Network	100%	0.0056	10.85%	10.27	0.1723
Convolutional Neural Network (with dense layers)	**85.43%**	**0.4188**	**93.67%**	**0.2041**	**0.0111**

Table 4: Result after training the models

After the training is completed ,CNN model shows decent results with a validation accuracy of 93.67 % for the predictions made.On the other hand,the validation accuracy of the DNN model is only 10.85% ,which is very low.

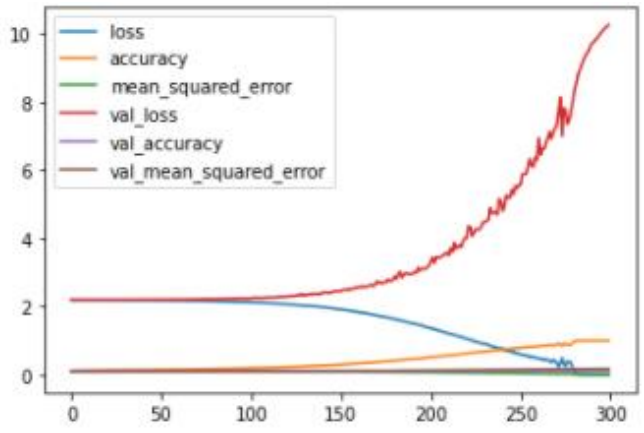


Figure 1: Training graph of DNN model

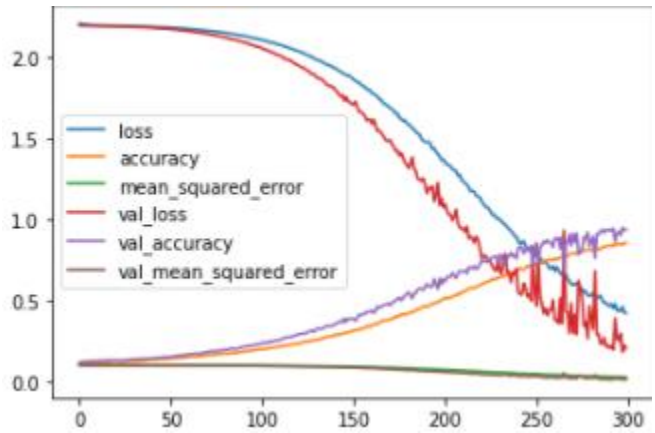


Figure 2: Trainng graph of CNN model

Moreover, looking at the training graphs above, the red line denotes the validation loss in both the graphs. As observed, the validation loss keeps increasing in the DNN model ,which indicates overfitting whereas in the case of CNN model(FIG 2),it decreases with training.

CONCLUSION AND FUTURE SCOPE

After the data is used to train both the models,it is clear that the CNN model(with dense layers) has much better results as compared to the DNN model as it has an accuracy and validation accuracy of 85.43% and 93.67% respectively.So,the model can be used to help students in deciding which courses should they do to further improve their skill set and be prepared in a better way for their placements and jobs.

The CNN model has a decent prediction accuracy but it can still be improved by a greater number of epochs and the training parameters can be experimented with.

With the appropriate dataset, a similar model can be made which can be used for students of all branches and not just only for students of Computer Science.

Further, the model can be deployed on a website using Web development and then it can be used by various colleges and universities for assisting their students.

REFERENCES

[1] Vairachilai , Vamshidharreddy, “Student’s Academic Performance Prediction Using Machine Learning Approach”, International Journal of Advanced Science and Technology Vol. 29, No. 9s, (2020), pp. 6731-6737, 2nd June 2020.

[2] Dinh Thi Ha ,Cu Nguyen Giap,Pham Thi To Loan,Nguyen Thi Lien Huong, “An Empirical Study for Student Academic Performance Prediction Using Machine Learning Techniques”, International Journal of Computer Science and Information Security (IJCSIS) , Vol. 18, No. 3, March 2020

[3] K. Sripath Roy , K.Roopkanth , V.Uday Teja , V.Bhavana, J.Priyanka, “Student Career Prediction Using Advanced Machine Learning Techniques”, International Journal of Engineering & Technology, Vol 7, No 2.20 (2018),April 2018

[4] Er.Parveen Kumar,Er.Pooja Sharma, “Artificial Neural Networks-A Study”, International Journal of Emerging Engineering Research and Technology, Volume 2,PP 143-148,Issue 2, May 2014

[5] Bendangnuksung, Dr. Prabu P, "Students' Performance Prediction Using Deep Neural Network",International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 2 (2018) pp. 1171-1176

[6] Mukesh Kumar, Yass Khudheir Salal , "Systematic Review of Predicting Student's Performance in Academics",International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-8 Issue-3, February 2019

[7] Ossama Abdel-Hamid, Li Deng, Dong Yu,"Exploring Convolutional Neural Network Structures and Optimization Techniques for Speech Recognition", INTERSPEECH 2013,August 25-29, 2013