RES MILITARIS Social Science Journal

UNDERSTUDIES' PERFORMANCE PREDICTION USING DEEP NEURAL NETWORK

MIDAKANTI PRADEEP REDDY, RAMA DEVI GUNDE, GOSU UPENDER

Dept of CSE,

Priyadarshini Institute of Science and Technology for Women Khammam

ABSTRACT

Educational Data Mining (EDM) and Deep Learning have attracted significant attention in recent years. In this study, we propose a Deep Neural Network (DNN) model aimed at predicting student performance, specifically classifying students into categories based on their likelihood of success or failure. The proposed model, compared with existing Artificial Intelligence (AI) algorithms using the same dataset, achieved an accuracy of up to 84.3%, outperforming other machine learning techniques. This paper presents a significant step forward in applying deep learning techniques in educational settings.

Keywords: Deep Neural Network (DNN), Deep Learning, Artificial Neural Network (ANN), Education Data Mining, Student Performance Prediction.

INTRODUCTION

The academic performance of students has always been a critical factor in determining their future career paths and the reputation of educational institutions. In recent years, there has been a growing interest in leveraging technology to enhance educational outcomes, with Educational Data Mining (EDM) emerging as a key discipline. EDM involves the extraction of significant data from educational settings, which can then be used to develop models for predicting student performance. The advent of Machine Learning (ML) techniques has further enhanced the ability to predict student performance based on various factors, such as background information and periodic assessments. These predictions can help identify students at risk of failing, allowing educators to intervene with appropriate measures. Furthermore, ML techniques can also be used to identify high-performing students, enabling institutions to offer scholarships and other forms of support.

Traditional ML algorithms, such as Decision Trees and Naive Bayes, have been widely used in EDM. However, these algorithms have limitations, particularly when dealing with continuous data. For instance, as Havan Agrawal [11] noted, the accuracy of Bayesian classification decreases when applied to continuous data, as these algorithms work better with discrete data. In contrast, Neural Networks (NN) tend to perform better when handling continuous data, making them a more suitable choice for predicting student performance. Deep Learning, a subset of ML, has emerged as a cutting-edge tool in AI research, with applications in various fields, including biometrics, natural language processing, and computer vision. Deep Learning techniques such as Deep Neural Networks (DNN), Recurrent Neural Networks (RNN), Convolutional Neural Networks (CNN), and Q-learning have shown remarkable success in these areas. In this paper, we propose a DNN model designed to predict student performance, specifically classifying students into pass or fail categories using logistic regression analysis. The model utilizes two hidden layers, with the first employing a ReLU activation function and the second utilizing a SoftMax activation function. The proposed model demonstrates its effectiveness in predicting students at risk of failing, achieving an accuracy of 85%.

LITERATURE REVIEW

The application of ML and DL techniques in educational settings is not new, but the field continues to evolve rapidly. Various studies have explored different approaches to predicting student performance, each contributing valuable insights into the strengths and limitations of these methods. Ioannis E. Livieris et al. [1] constructed an Artificial Neural Network (ANN) classifier to predict student performance in mathematics. Their experiments revealed that a modified spectral Perry-trained ANN outperformed other classifiers, demonstrating the potential of neural networks in educational data mining. S. Kotsiantis et al. [2] conducted a pioneering study on dropout prediction in distance learning systems using ML techniques. Their research played a significant role in shaping the field of educational data mining by demonstrating the effectiveness of ML techniques in academic environments. They utilized demographic data and various academic metrics to predict student dropouts, setting the stage for future studies in this area.

Moucary et al. [3] applied a hybrid approach combining K-Means clustering and ANN for students pursuing higher education in a foreign language. This approach allowed them to predict student performance and group students into clusters, providing educators with a powerful tool to identify students' capabilities at an early stage. The clustering method helped tailor educational interventions to students' specific needs. Hongsuk et al. [4] proposed a DNN model for predicting traffic conditions based on a traffic performance index. Their model, which employed a three-layer DNN, achieved nearly 100% accuracy in identifying congested and non-congested traffic conditions. Although the application was different, their work highlighted the effectiveness of DNNs in classification tasks, which is relevant to the prediction of student performance.

Amrieh et al. [12] proposed a model for predicting student performance based on data mining techniques, focusing on student behavior features. Their model was evaluated using three different classifiers: Naive Bayes, ANN, and Decision Tree. By employing ensemble methods such as Random Forest, Bagging, and Boosting, they were able to improve the classifier's performance significantly. Their model achieved up to 25.8% improvement in accuracy after using ensemble methods, demonstrating the importance of feature selection and model optimization in educational data mining. These studies illustrate the potential of various ML and DL techniques in predicting student performance. However, there is still room for improvement, particularly in addressing the limitations of existing models, such as their dependence on large datasets and the challenges of overfitting and underfitting.

DEEP NEURAL NETWORK (DNN)

Deep Learning techniques aim to learn feature representations at multiple levels of abstraction, enabling models to capture complex patterns in data. A DNN is a type of neural network model that consists of multiple hidden layers between the input and output layers. These hidden layers are composed of neurons, which are analogous to the neurons in the human brain. A neuron in a DNN is a nonlinear function that maps input vectors $\{I_1, \ldots, I_n\}$ to an output Y through a weighted vector $\{w_1, \ldots, w_n\}$ and an activation function f. The objective of the model is to optimize the weights w such that the squared loss error is minimized. This optimization is typically achieved using stochastic gradient descent (SGD), which iteratively updates the weight vector to guide the model toward the minimum gradient of the loss function.

An epoch in DNN training consists of one complete forward and backward pass through the dataset. During each epoch, the model's parameters are updated to minimize the loss function. The training process involves iterating through the data multiple times, refining the model's weights and biases with each iteration. DNNs are particularly well-suited for tasks that involve large datasets and complex, high-dimensional data. They have demonstrated exceptional performance in a wide range of applications, including image recognition, speech processing, and natural language understanding. In this study, we explore the application of DNNs to predict student performance in an educational setting, leveraging the model's ability to learn from complex patterns in the data.

METHODOLOGY

In this section, we describe the methodology used to develop the DNN model for predicting student performance. The process includes data preprocessing, model architecture design, and the training and evaluation of the model. The dataset used in this study was sourced from the Kalboard 360 Learning Management System, which is available on Kaggle [https://www.kaggle.com/aljarah/xAPI-Edu-Data]. The dataset comprises 500 records of students, each containing 16 distinct features related to their academic performance and background. The dataset includes three classes based on students' performance, categorized by their numerical interval values. The features include demographic information, academic performance metrics, and behavioral data.

Before training the DNN model, it was essential to preprocess the data to ensure its quality and suitability for model training. Data preprocessing involved several steps, including data cleaning, transformation, and reduction. The dataset contained 20 missing values across various features. These missing values were removed, reducing the dataset to 480 records. This step was crucial to ensure that the model received complete and accurate information for training. The dataset included several nominal features, such as Gender, Relation, Semester, Parent Answering Survey, Parent School Satisfaction, and Student Absence Days. These features were transformed into binary data ('0' and '1'). Additionally, other nominal features like Nationality, Place of Birth, Stage ID, Grade ID, Section ID, and Topic were converted into numerical data.

The proposed DNN model was built using Python 3 and TensorFlow 1.3.0. Python was chosen due to its widespread use in scientific computing and its extensive support for ML and DL libraries. TensorFlow, an open-source library for numerical computation using data flow graphs, was used to implement the DNN model. The architecture of the DNN model includes the following components: The input layer receives the preprocessed features from the dataset. The model includes two hidden layers, each consisting of 300 neurons. The first hidden layer uses the ReLU (Rectified Linear Unit) activation function, which is defined as $f(x) = \max(0, x)$. The second hidden layer uses the SoftMax activation function, which converts the input into a probability distribution, making it suitable for classification tasks. The output layer produces the final prediction, classifying students into pass or fail categories based on their performance. The training process involved dividing the dataset into training and testing sets, with a 3:1 ratio. The training set was used to train the DNN model, while the testing set was reserved for evaluating the model's performance. The training process began by initializing the weights w and biases b for each layer. The weights were updated using the Adam optimizer, a variant of SGD that adapts the learning rate based on the first and second moments of the gradients. The Adam optimizer is well-suited for training DNNs, as it helps achieve faster convergence and better performance. The model was trained for 50 epochs, with each epoch consisting of a forward pass (calculating predictions) and a backward pass (updating weights based on the error). The error was computed using the cross-entropy cost function, which measures the difference between the predicted and actual outputs. The goal was to minimize the cost function, thereby improving the model's accuracy.

The model's performance was evaluated using two primary metrics: accuracy and cost function. Accuracy measures the proportion of correctly classified instances, while the cost function evaluates the error between the predicted and actual outputs. The initial accuracy of the model was 29.8%. However, after 50 epochs of training, the model's accuracy increased to 84.3%. The cost function started with a high value of 15560 but gradually decreased as the model optimized its weights. The final model demonstrated a significant improvement in accuracy, outperforming other ML algorithms, such as Decision Trees, Naive Bayes, and ANN, on the same dataset.

Experimental Results and Analysis

The experimental process involved testing the DNN model on a set of student records to evaluate its performance in predicting academic outcomes. The experiments were conducted on an Ubuntu 16.04 operating system with a configuration of 8GB RAM and 4 Intel cores. The model was implemented using Python 3 and TensorFlow, with visualization tools like TensorBoard and Matplotlib used to monitor the model's progress. The model took approximately 5 minutes to execute, reflecting the computational efficiency of the DNN architecture. Despite the relatively small dataset, the DNN model demonstrated its ability to learn complex patterns, achieving a final accuracy of 84.3%. During the training process, the cost function showed a steady decrease, indicating that the model was successfully minimizing the error. The initial cost function value was 15560, which decreased significantly over the course of training. However, there was a slight increase in the cost value during the 17th iteration due to ineffective gradient updates, but the cost continued to decrease from the 34th iteration onwards.

The training was stopped after 50 epochs, as further iterations led to an increase in the cost function, suggesting the risk of overfitting. Overfitting occurs when the model becomes too complex, capturing noise in the training data rather than the underlying pattern, which can lead to poor generalization on new data. A comparative analysis was conducted to evaluate the performance of the proposed DNN model against other ML algorithms. The comparison was made with the model proposed by Amrieh et al. [12], which utilized Decision Tree, Naive Bayes, and ANN classifiers on the same dataset.

Table: IV

Classification Method Comparison	
Classifier	Accuracy
Decision Tree (J48)	82.2
Artificial Neural Netork (ANN)	80.0
Naive Bayes (NB)	80.0
Proposed Model (DNN)	84.3

The comparison was visualized using a whisker plot, which showed the spread of accuracy scores across 10-fold cross-validation for each algorithm. The results indicated that the DNN model outperformed the other algorithms, even with a smaller dataset, demonstrating the model's robustness and accuracy in predicting student performance. Out of 120 students in the testing set, 19 students were incorrectly classified by the DNN model. Despite this, the model's overall performance was superior to other classification techniques, making it a reliable tool for predicting student outcomes.

CONCLUSION AND FUTURE WORK

This paper presents a DNN model for predicting student performance, marking the first application of deep learning in educational data mining for this specific purpose. The proposed model achieved an accuracy of 84.3%, outperforming traditional ML algorithms on the same dataset. The study demonstrates that DNNs can effectively predict student performance, even with limited data, provided that the model is carefully tuned and optimized. The ability of DNNs to learn complex patterns and representations makes them a powerful tool for educational data mining. However, the study also highlights the importance of understanding the dataset and the model's architecture. Overfitting and underfitting are common challenges in deep learning, and careful attention must be paid to the number of epochs and the size of the hidden layers to ensure optimal performance. Future work could focus on expanding the dataset and exploring different DNN architectures to further improve accuracy. Additionally, integrating more features, such as behavioral data and extracurricular activities, could provide a more comprehensive view of student performance. Overall, the proposed DNN model offers a promising approach for educational institutions to predict student outcomes and intervene early to support at-risk students, ultimately improving academic success rates.

REFERENCES

1. Livieris, et al. (2012): Predicting students' performance using artificial neural networks. 8th PanHellenic Conference with International Participation Information and Communication Technologies, pp.321-328.

2. S. Kotsiantis, et al. (2003): Preventing student dropout in distance learning systems using machine learning techniques. Applied Artificial Intelligence, 18(5), pp.411-426.
3. Moucary, C.E., Khair, M., & Zakhem, W. (2011): Improving student's performance using data clustering and neural networks in foreign-language based higher education. The Research Bulletin of Jordan ACM, 2(3), pp.27-34.
4. Yi, H., Jung, H., & Bae, S. (2017, February): Deep Neural Networks for traffic flow prediction. In Big Data and Smart Computing (BigComp), 2017 IEEE International Conference on (pp. 328-331). IEEE.
5. LeCun, Y., Bengio, Y., & Hinton, G. (2015): Deep learning. Nature, 521(7553), pp.436-444.
6. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., & Darrell, T. (2014, November): Caffe: Convolutional architecture for fast feature embedding. In Proceedings of the 22nd ACM international conference on Multimedia (pp.675-678).
7. Dahl, G.E., Yu, D., Deng, L., & Acero, A. (2012): Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. IEEE Transactions on audio, speech, and language processing, 20(1), pp.30-42.
8. Collobert, R., & Weston, J. (2008, July): A unified architecture for natural language processing: Deep neural networks with multitask learning. In Proceedings of the 25th international conference on Machine learning (pp.160-167).
9. Koutina, M., & Kermanidis, K.L. (2011): Predicting postgraduate students' performance using machine learning techniques. In Artificial Intelligence Applications and Innovations (pp.159-168). Springer, Berlin, Heidelberg.
10. Saini, P., & Jain, A.K. (2013): Prediction using Classification Technique for the Students' Enrollment Process in Higher Educational Institutions. International Journal of Computer Applications, 84(14).
11. Agrawal, H., & Mavani, H. (2015): In Student Performance Prediction using Machine Learning. International Journal of Engineering Research and Technology.
12. Amrieh, E.A., Hamtini, T., & Aljarah, I. (2016): Mining Educational Data to Predict Student's academic Performance using Ensemble Methods. International Journal of Database Theory and Application, 9(8), pp.119-136.
13. Gorr, W.L., Nagin, D., & Szczypula, J. (1994): Comparative study of artificial neural network and statistical models for predicting student grade point averages. International Journal of Forecasting, 10(1), pp.17-34.
14. Li, J., Zhao, R., Huang, J.T., & Gong, Y. (2014): Learning small-size DNN with output-distribution-based criteria. In Fifteenth Annual Conference of the International Speech Communication Association.