



Course Recommendation with Deep Learning Approach

Tran Thanh Dien^(✉), Luu Hoai-Sang, Nguyen Thanh-Hai,
and Nguyen Thai-Nghe^(✉)

Can Tho University, Can Tho, Vietnam
{thanhdien, lhsang, nthai, ntnghe}@ctu.edu.vn

Abstract. Course selection is a crucial task which may affect greatly on student performance. Because of poor performances, numerous students have been receiving formal warnings and expulsions from universities. Clearly, a good strategy for study progress which can come from course recommendation methods really holds an important role to obtain a good study performance. In addition, early warnings that release on challenging courses enable students to prepare better for such courses. The current course recommendation systems are usually conducted from marks prediction and factor analysis on marks of courses based on advancements of machine learning approaches. In this study, we propose a course recommender system by using deep learning techniques with MultiLayer Perceptron and pre-processing methods. The prediction tasks are performed on approximately four million of mark records at Can Tho University, Vietnam to provide recommendations on course selection to students. The proposed method reveals promising results and is expected to apply in practical cases.

Keywords: Course selection · Deep learning · MultiLayer Perceptron · Student performance prediction · Course recommendation

1 Introduction

Student performance is an important task of higher educational institutions because it is a criteria for high quality universities that are based on excellent profile of their academic achievements. According to [1], student performance can be obtained by measuring the learning assessment and curriculum. However, most of the studies mentioned that student performance based on the measurement of students' success [2].

Several situations of student performance in the universities are released to student and family's student to warn in the case that students got poor performance. Taking an example from Can Tho University, in the first semester of the academic year 2018–2019, there were more than 800 released warnings for one-poor-performance semester and more than 100 for the two-consecutive-poor-performance semesters cases. These numbers are rising with 986 and 196

respectively for the academic year 2019–2020¹. One of the main reasons for the students' poor performance is that they have not selected appropriate courses to their competencies. These results in extension of learning term and increase of cost for their families, higher educational institutions and society as well. Therefore, predicting students' performance is an important research topic in exploiting educational data, which is of interest to many researchers [3].

In order to students to obtain the best academic performance, recommender systems is needed for course selection to predict the best courses that should be selected by students. There are the existing recommender systems for selecting courses. For example, data mining is one of the most popular approaches to be widely applied in educational area. One of the most popular techniques to predict student performance is classification. There are several algorithms used for classification task such as Decision Tree, Artificial Neural Networks, Naive Bayes, K-Nearest Neighbor and Support Vector Machines [2]. However, the existing researches are primarily based on learning results of previous semesters to predict student performance of next semester or Current Grade Point Average (GPA), but do not analyze additional factors such as English entrance testing grades, activity incentive grades, etc. that affect their performance. Moreover, the researchers have not sufficiently compared among techniques, especially deep learning techniques with other traditionally machine learning techniques.

This study proposes a course selection recommendation system by using deep learning techniques [4], especially the multi-layer perceptron (MLP) to build a student's performance prediction model for predicting student performance in next semesters based on the course's achievement results of the previous semesters. In addition, in order to improve the predictive results, we also consider other additional factors such as entrance English testing grades, activity incentive grades etc. for the proposed model. Moreover, a comparison between deep learning techniques and traditionally machine learning ones is also conducted. Experiential data is collected from the student information system of a Vietnamese multidisciplinary university. The results show that the proposed model provides rather accurate prediction and it can be applied in practical other cases.

2 Related Work

Course recommendation is really important to build a good study strategy for students. Mark prediction tasks are required to provide valuable advices for course recommendation. Numerous studies have attempted to propose efficient methods for student performance. The authors in [5], introduced an approach implementing Tensor Factorization (TF) to predict student performance. The proposed method enabled us to personalize the prediction for specific student. The authors reported promising experiential results on two large datasets.

¹ Can Tho University, 2020. Management Information System, accessed on 12/5/2020.
Available from <https://htql.ctu.edu.vn/>.

Recently, open source libraries have been more used for predictive work. [6] developed a student performance prediction system using the open source recommendation system called *MyMediaLite*. For the grade databases collected from the academic management system of a university, the authors proposed using Biased Matrix Factorization (*BMF*) technique to predict the learning results. This results can help students choose more appropriate courses. The authors in [7] proposed a novel approach which uses recommender system techniques for educational data mining, especially for predicting student performance. For validating this approach, the recommender system techniques were compared with traditional regression methods such as linear regression by using educational data for intelligent tutoring systems. The experimental results showed that the proposed approach can improve prediction results.

The work in [8] presented an approach to boost student performance prediction in interactive online question pools through considering student interaction features and the similarity between questions. The proposed approach evaluated on the dataset from a real-world interactive question pool using four typical machine learning models. The result showed that the approach can achieve a much higher accuracy for student performance prediction in interactive online question pools than the traditional way using the statistical features in various models.

The authors in [9] evaluated exhaustively the prediction performance based on all possible combinations of four types of attributes including behavioral features, demographic features, academic background, and parent participation. Using support vector machine and feature selection, the authors concluded that the behavioral feature is so crucial because of the optimal prediction performance. In [10], the authors proposed a method of predicting student performance in computing majors. This method is based on collaborative filtering using enhanced similarity and yields personalized predictions of student grades in courses required for each computing major. Prediction accuracy is enhanced by analyzing computing major-specific course characteristics, such as core courses, course prerequisites, and course levels.

The authors with the research in [11] proposed a methodology in which the process of data collection and pre-processing is carried out, then the grouping of students with similar patterns of academic performance was carried out. The authors selected the most appropriate supervised learning algorithm. The experimental results showed the effectiveness of machine learning techniques to predict the performance of students. The authors in [12] used the efficacy of Matrix Factorization as a technique for solving the prediction problem. The study uses Singular Value Decomposition (SVD), a Matrix Factorization technique that has been successfully used in recommender systems. The performance of the technique was benchmarked against the use of student and course average marks as predictors of performance. The results obtained suggests that Matrix Factorization performs better than both benchmarks. The work in [13] proposed a direct search of the optimal value of latent factors where the calculation for each number of latent factors is driven by a metaheuristic that select, at the same time,

the optimal values of learning rate and regularization factor. Using this method, authors can determine the best number of latent factors to be applied in further predictions for the similar databases.

Another study in [14] introduced a supervised content-aware matrix factorization for mutual reinforcement of academic performance prediction and library book recommendation. The proposed model was evaluated on three consecutive years of book-loan history and cumulative grade point average of 13047 undergraduate students in one university. The results showed that the proposed model outperforms the competing baselines on both tasks, and that academic performance not only is predictable from the book-loan history but also improves the recommendation of library books for students.

An optional course recommendation system based on score prediction was introduced in [15] using a novel cross-user-domain collaborative filtering algorithm to accurately predict the score of the optional course for each student by using the course score distribution of the most similar senior students. The experimental results showed that the proposed method is able to accurately recommend optional courses to students who will achieve relatively high scores. Another work of [16] proposed a method to predict student performance using various deep learning techniques. Several techniques for data pre-processing used such as Quantile Transforms, MinMax Scaler before fetching them into well-known deep learning models such as Long Short Term Memory and Convolutional Neural Networks to do prediction tasks. The experiential results showed that the proposed method provides good prediction results, especially using data transformation.

Some studies also introduced methods for course recommendation. [17] proposed a collaborative recommender system that recommends university elective courses to students by exploiting courses that other similar students had taken. The system used an association rules mining algorithm to discover patterns between courses. The experiments were conducted with real datasets to assess the overall performance of the proposed approach.

In order to support students choose their subjects as per their capability, [18] used data mining and natural language processing techniques that helps in conversion of human-readable format to machine-readable format, both of which are vastly emerging fields to propose a collaborative recommendation system. In addition, [19] proposed a novel course recommendation system based collaborative filtering considering the case of repeating a course and students' grades in the course for each repetition. The authors experimented different Ordered Weighted Averaging operators which aggregates grades for each student's repeated courses to enhance the recommendation quality. [20] developed an optional course recommendation system based on score prediction. A novel cross-user-domain collaborative filtering algorithm is designed to accurately predict the score of the optional course for each student by using the course score distribution of the most similar senior students. After generating the predicted scores of all optional courses, the top t optional courses with the highest predicted scores without time conflict will be recommended to the student.

Based on the previous research results, we propose an approach of deep learning techniques using a MLP to build a student's performance prediction model. Beside course grades, additional factors such as entrance English testing grades, activity incentive grades, etc. also considered to improve the proposed prediction model.

3 Proposed Approach

General framework of the prediction models in the proposed approach is described in detail in Fig. 1. First, we collect real data sets at the Student Management System of a university, then data is pre-processed to remove noise, redundant attributes, etc. Traditional approaches usually used full dataset to build one prediction model (presented in the bottom of Fig. 1) to predict all of the students (denoted as **MLP**), however, we realize that this approach does not fit to the data since we should not use a “very good performance student” to predict a “poor performance student” or vice versa. In this work, we propose using four prediction models for four group of students which have similar performance based on their marks (as presented in the top of Fig. 1). The proposed approach is called **GroupMLP**.

3.1 Data Pre-processing

For evaluating the proposed model, we have collected real data at Can Tho University, Vietnam; however, the model can be used for other case studies such as other universities, schools, colleges as well. The collected data relates to students, courses, marks, and other information from the year 2007 to 2019 with 3,828,879 records, 4,699 courses (subjects), and 83,993 students. Data distributions are described in Fig. 2.

The considered data set is collected from Student Management system and pre-processed as described in Algorithm 1.

After carefully analyzing the data, we have selected the input attributes for learning model as described in Table 1. This selection based on pre-experimental results and previous analysis in predicting student performance [6, 21].

3.2 Deep Learning Architecture

The proposed model is presented in Fig. 3 that was used in the work of [22]. The input attributes are selected from Table 1 and the output (prediction) of the model having eight classes which are the four-grading system (i.e., the output belongs to $[0, 1, 1.5, 2, 2.5, 3, 3.5, 4]$).

The proposed MLP architecture includes 6 layers. The first layer contains 18 input features while the last layer consists of 1 output exhibiting the predicted mark. The first four hidden layers contain 256 neurons, but they are followed by different activation functions. The first hidden and the fourth hidden layer use Rectified Linear Unit (ReLU) while other hidden layers are followed by Sigmoid

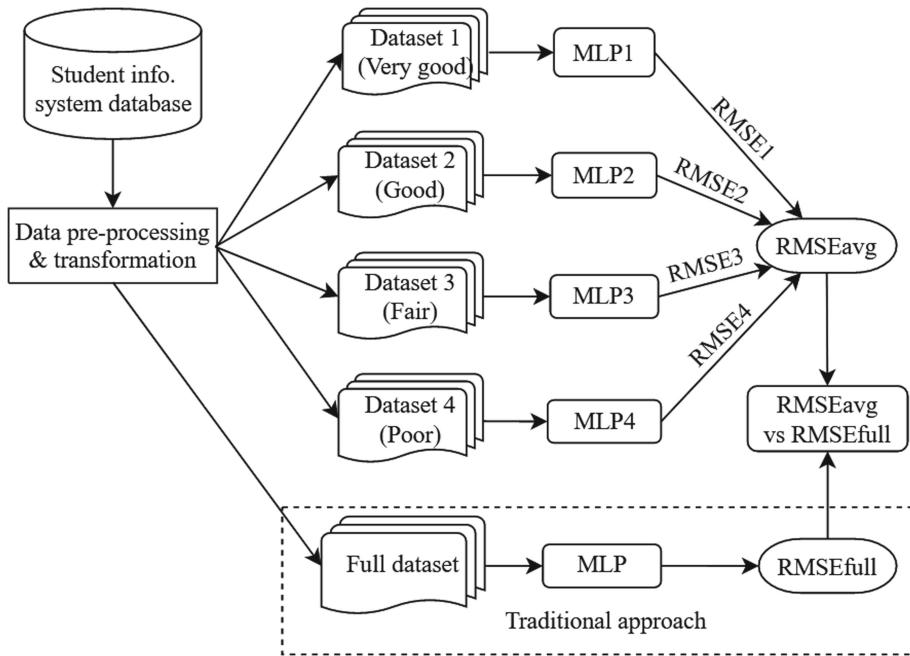


Fig. 1. General diagram of the proposed approach

■ Train ■■■ Test ■■■■■ Total

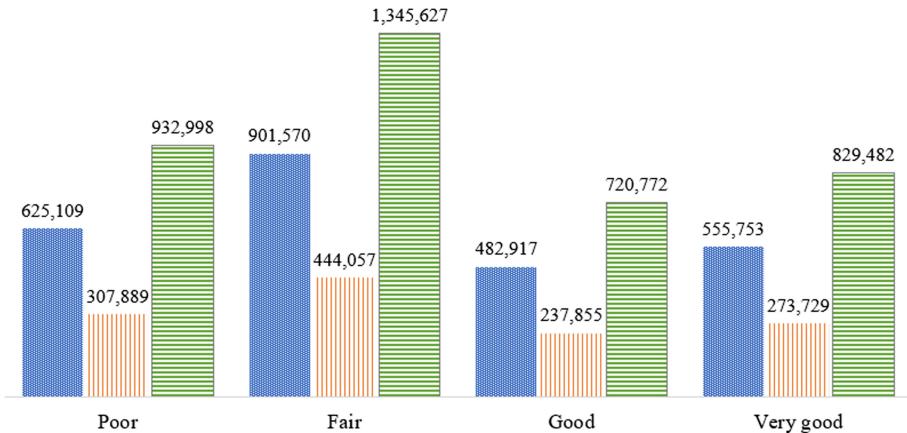


Fig. 2. Data set distribution by grading score

function. Before producing predicted score, we implement the fifth hidden layer containing 8 nodes using ReLU.

Algorithm 1. Algorithm for pre-processing and filtering features from student management system

Begin

- Step 1:** Redundant attributes such as Student Name, Course Name, Lecturer Name, etc. are eliminated from the original set of features collected from the Can Tho University's student management system.
- Step 2:** For each student, mark entries which did not contain a specific value (for example, null or empty values) or was exemption courses, etc. were removed.
- Step 3:** Transform features which is text type to numeric values.

End

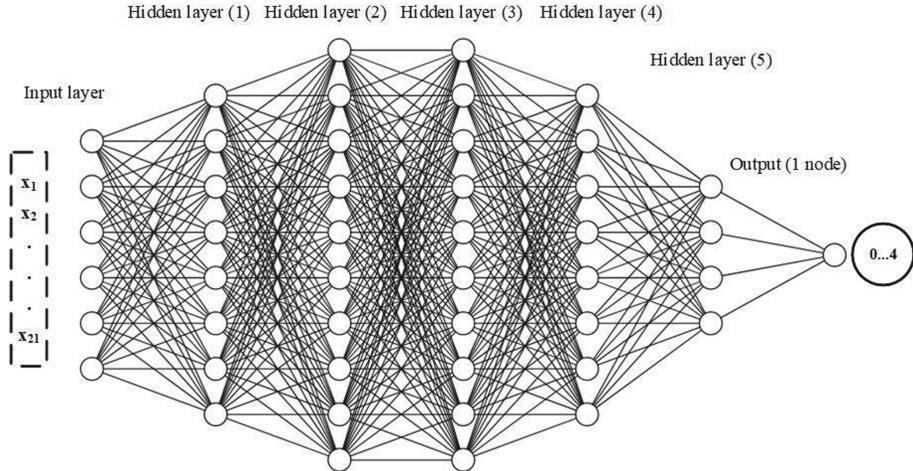


Fig. 3. The proposed MLP architecture [22].

The Sigmoid function [23] usually appears in the output layers of Deep learning architectures. It transforms the input values which lie in the domain \mathbb{R} to outputs have the domain in $[0, 1]$. The Sigmoid function is also called “squashing” because this function squashes any input in the range of $(-\infty, \infty)$ to range of $[0, 1]$. When we shifted to gradient based learning, the Sigmoid was considered as a natural selection due to its smooth and differentiable approximation to a thresholding unit. The Sigmoid function is given by the formula:

$$\text{Sigmoid}(x) = \frac{1}{1 + e^{-x}} \quad (1)$$

where, x denotes data after being computed by the preceded neural layer.

Table 1. Input attributes

No.	Attribute name	Description
1	studentID	Student ID
2	Gender	Student gender
3	Hometown	Student hometown
4	EntranceMark1	Entrance mark for Course 1
5	EntranceMark2	Entrance mark for Course 2
6	EntranceMark3	Entrance mark for Course 3 (The students take an entrance test with 3 courses)
7	FoS	Field of study
8	Faculty	Student belongs to a faculty (e.g. IT, Agriculture,...)
9	TrainingNumber	Course of the student, e.g. 2000–2004, 2002–2006,
10	Semester	
11	GPA	Current Grade Point Average
12	CGPA	Cumulative Grade Point Average
13	TotalCredits	Total of taken credits by student
14	AVGMark	Average of previous marks
15	LecturerID	The lecturer who taught the course
16	CourseID	ID of the course
17	CreditNo	Number of credit of the course
18	MarkEntryDate	The date when the mark is entered
19	BSC1	Basic English Course 1
20	BSC2	Basic English Course 2
21	BSC3	Basic English Course 3

Another activation namely, ReLU [24], is also implemented in our architecture. ReLU follows the formula:

$$f(x) = \max(0, x) \quad (2)$$

where, x denotes data after being processed by the preceded neural layer.

ReLU is the most widely used activation function for deep learning architectures with state-of-the-art results to date. ReLU helps models to produce the better performance and generalization in deep learning compared to the Sigmoid and Tanh activation functions. It represents a nearly linear function, so this activation function preserves the properties of linear models that made them easy to optimize, with gradient-descent method [25, 26].

In order to reduce overfitting issues, dropout technique is deployed with a rate of 0.015. In addition, we also consider using Early Stopping with a patience epoch of 5. If the loss cannot be improved after 5 consecutive epochs, the learning will be stopped. Otherwise, the learning will be continued to run to 500 epochs. The network is implemented with Adam optimizer function, use a batch size of 255 and a default learning rate of 0.001.

4 Evaluation

To evaluate results of the proposed model, we describe some baselines and state-of-the-art for comparison and two popular metrics for measuring the performance. We have not applied cross validation since the data are in order, i.e., using the students' marks of previous semesters to predict the marks of current semester.

4.1 Baseline Methods

In this work, to compare with other methods, we have used two baselines which are User Average and Item Average. Furthermore, we also compared with Collaborative Filtering methods since previous works [5, 27] showed that using Collaborative Filtering such as state-of-the-art Matrix Factorization [28] provided very good results in predicting student performance.

Let denote u as the student, i as the course, and r as the mark of the student on that course. The User Average method, which can be known as Student Average in this work, generates prediction mark (\hat{r}_{ui}) for student u on course i by using formula 3.

$$\hat{r}_{ui} = \frac{\sum_{(u',i,r) \in \mathcal{D}^{train} | u' = u} r}{|\{(u',i,r) \in \mathcal{D}^{train} | u' = u\}|} \quad (3)$$

The Item Average is Course Average in this work, which predicts the mark for the student u on course i by using formula 4.

$$\hat{r}_{ui} = \frac{\sum_{(u,i',r) \in \mathcal{D}^{train} | i' = i} r}{|\{(u,i',r) \in \mathcal{D}^{train} | i' = i\}|} \quad (4)$$

Matrix Factorization is a well-known method in recommender systems, it decomposes a matrix X (each row of X is a user/student, each column is an item/course, and each element is a mark for the student on that course, respectively) to two small matrices W and H such that we can reconstruct X from these two matrices

$$X \approx WH^T$$

where $W \in \mathbb{R}^{|U| \times K}$; $H \in \mathbb{R}^{|I| \times K}$; K is number of latent factors, $K \ll |U|, K \ll |I|$. The latent factors W and H can be obtained from optimizing the function:

$$\mathcal{O}^{MF} = \sum_{(u,i) \in \mathcal{D}^{train}} \left(r_{ui} - \sum_{k=1}^K w_{uk} h_{ik} \right)^2 + \lambda \cdot (\|W\|_F^2 + \|H\|_F^2)$$

$\lambda \in (0..1)$ is a regularization and $\|\cdot\|_F$ is the Frobenius norm.

One benefit of the Matrix Factorization approach is its flexibility in dealing with various data aspects. However, the variation in the rating values are due to

effects associated with either users or items, known as biases. Thus, researchers extended the previous work as Biased Matrix Factorization.

Biased matrix factorization [31], an improvement of MF, models the characteristics of each user and each item and the global tendency that are independent of user-item interactions.

At that time, the optimizing the function described as follows:

$$\mathcal{O}^{BMF} = \sum_{(u,i) \in \mathcal{D}^{train}} \left(r_{ui} - \mu - b_u - b_i - \sum_{k=1}^K w_{uk} h_{ik} \right)^2 + \lambda \cdot (\|W\|_F^2 + \|H\|_F^2)$$

where:

- μ is the global average (average performance of all students and tasks in D^{train}) as shown in Eq. 5.
- b_u (exhibited in Eq. 6) is student bias (average performance of student u deviated from the global average).
- b_i (Eq. 7) is mark bias (average performance on mark i deviated from the global average).

$$\mu = \frac{\sum_{(u,i,r \in D^{train})} r}{|D^{train}|} \quad (5)$$

$$b_u = \frac{\sum_{(u',i,r \in D^{train})|u'=u} (r - \mu)}{|\{(u',i,r \in D^{train})|u'=u\}|} \quad (6)$$

$$b_i = \frac{\sum_{(u,i',r \in D^{train})|i'=i} (r - \mu)}{|\{(u,i',r \in D^{train})|i'=i\}|} \quad (7)$$

Details of these methods are described in [5, 27–31].

4.2 Evaluation Metrics

The root mean squared error (RMSE) and mean absolute error (MAE) are used to evaluate the models. They are calculated by Eqs. (8) and (9), respectively.

$$\sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (8)$$

$$\frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (9)$$

where, y_i is the true value, and \hat{y}_i is the predicted value.

4.3 Experimental Results

Experimental results of MAE and RMSE are presented in Fig. 4. The GroupMLP presents for using four different models for four groups of student performances and the MLP means using one model to predict the result of all the students, as mentioned in Fig. 1. Clearly, by using different prediction models for different student performances, the results are significantly improved.

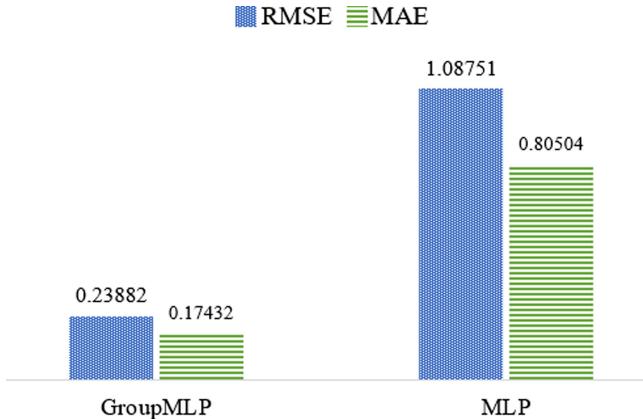


Fig. 4. RMSE and MAE comparisons between GroupMLP and MLP

Figure 5 presents the comparison results between the proposed GroupMLP and other baselines in Recommender Systems. This result shows that the GroupMLP works better than other baselines, however, using train (67%)/test(33%) split by time, the recommender system techniques may suffer from the “cold-start problem”, i.e., new-users and new-items may happen in the test set.

Figure 6 presents the results for each model in the group (as presented in Fig. 1). The results show lower errors where levels of Very Good and Good exhibit promising performance. However, with lower levels of marks such as Fair and Poor reveals high errors both in MAE and RMSE in the prediction.

Moreover, while analyzing the results, we obtain that the English skill of the students is very important attribute. For example, without using English marks (the *BSC1*, *BSC2*, *BSC3* in Table 1) the MAE of the MLP model is 0.24564, while using these attributes with english courses, the MAE is dropped to 0.23882 as shown in Table 2. This result implies that for the students having better English, they can read learning resources (mostly in English) better.

Deep learning approach may take more times for training the models, however, this is not a problem since models are trained every one or two times in the semester when the marks are changed in mid-term and final examination.

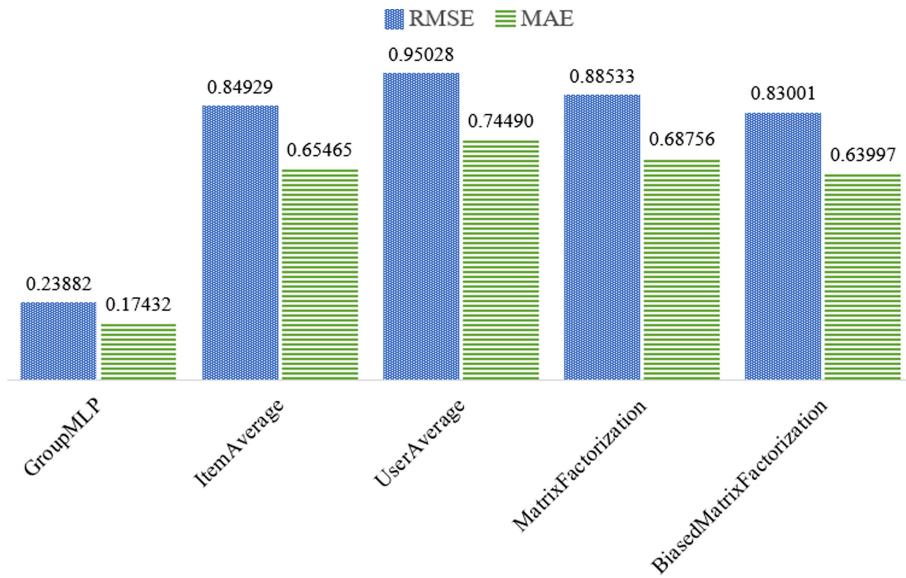


Fig. 5. RMSE and MAE comparisons between GroupMLP and baselines

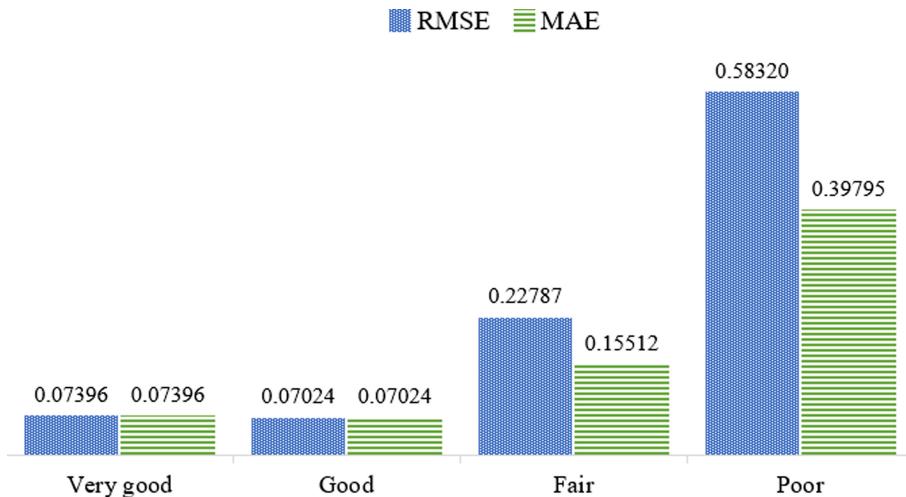


Fig. 6. Performance comparison with mark levels in MAE and RMSE

Table 2. Feature analysis on English skill

	MAE	RMSE
Without using English marks	0.24564	0.18064
Using English marks	0.23882	0.17432

4.4 Model for Early Warning and Course Recommendation

After evaluating the model, we have attempted to apply the proposed method for the real case study as in Fig. 7. Given the student information (or a list of students) and the course information as described in Table 1, the model can generate prediction results for the student on that course. We can use the result in two cases:

- **Case 1:** If the prediction result is less than 1.0 (or other selected threshold in the four-grading system, from 0 to 4), that would be the warning case.
- **Case 2:** If the prediction result is more than 3.25 (or other selected threshold) for the elective/selection course, that would be the recommendation course for the student to select.

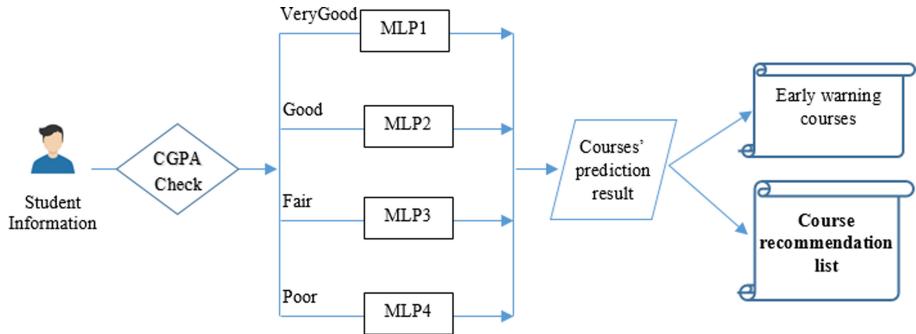


Fig. 7. The proposed early warning and course recommendation model for students

5 Conclusion

In this study, we proposed a method to predict student performance using deep learning techniques for prediction tasks on the data collected from a Vietnamese multidisciplinary university's information system. We analyze and propose some techniques for data pre-processing before fetching them into a MLP to do prediction tasks. The proposed method provides good prediction results and is expected to apply in practical cases. Using these results, we can help the educational managers or the academic advisors and the students to know early warning results so that the students can have better plan for studying. Moreover, evaluating on various training courses to help the managers to propose appropriate policies.

We continue to perform experiments on other published data sets and to change the model settings for better performance. Further research should investigate more on groups based on student performance to support better students' course selection and enhance the prediction tasks performance.

References

1. Mat, U., Bunyamin, N., Arsal, P., Kassim, R.: An overview of using academic analytics to predict and improve students' achievement: a proposed proactive intelligent intervention. In: 2013 IEEE 5th Conference on Engineering Education (ICEED) (2013)
2. Shahiri, A., Husain, W., Rashid, N.: A review on predicting student's performance using data mining techniques. *Procedia Comput. Sci.* **72**, 414–422 (2015)
3. Guo, B., Zhang, R., Xu, G., Shi, C., Yang, L.: Predicting students performance in educational data mining. In: 2015 International Symposium on Educational Technology (ISET) (2015)
4. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. *Nature* **521**, 436–444 (2015). <https://doi.org/10.1038/nature14539>
5. Thai-Nghe, N., Horvath, T. and Schmidt-Thieme, L.: Factorization models for forecasting student performance. In: Educational Data Mining (2011)
6. Huynh-Ly, T.-N., Thai-Nghe, N.: A system for predicting students's course result using a free recommender system library of MyMediaLite (in Vietnamese). In: Information Technology Conference (2013)
7. Thai-Nghe, N., Drumond, L., Krohn-Grimberghe, A., Schmidt-Thieme, L.: Recommender system for predicting student performance. *Procedia Comput. Sci.* **1**, 2811–2819 (2010)
8. Wei, H., Li, H., Xia, M., Wang, Y., Qu, H.: Predicting student performance in interactive online question pools using mouse interaction features. In: Proceedings of the Tenth International Conference on Learning Analytics and Knowledge (2020)
9. Hirokawa, S.: Key attribute for predicting student academic performance. In: Proceedings of the 10th International Conference on Education Technology and Computers - ICETC 2018, pp. 308–313 (2018)
10. Park, Y.: Predicting personalized student performance in computing-related majors via collaborative filtering. In: Proceedings of the 19th Annual SIG Conference on Information Technology Education (2018)
11. Buenaño-Fernández, D., Gil, D., Luján-Mora, S.: Application of machine learning in predicting performance for computer engineering students: a case study. *Sustainability* **11**, 2833 (2019)
12. Jembere, E., Rawatlal, R., Pillay, A.: Matrix factorisation for predicting student performance. In: 2017 7th World Engineering Education Forum (WEEF) (2017)
13. Duran-Dominguez, A., Gomez-Pulido, J., Rodriguez-Lozano, D., Pajuelo-Holguera, F.: Selecting latent factors when predicting student performance in online campus by using recommender systems. In: 2018 13th Iberian Conference on Information Systems and Technologies (CISTI) (2018)
14. Lian, D.-F., Liu, Q.: Jointly recommending library books and predicting academic performance: a mutual reinforcement perspective. *J. Comput. Sci. Technol.* **33**(4), 654–667 (2018). <https://doi.org/10.1007/s11390-018-1847-y>
15. Huang, L., Wang, C., Chao, H., Lai, J., Yu, P.: A score prediction approach for optional course recommendation via cross-user-domain collaborative filtering. *IEEE Access* **7**, 19550–19563 (2019)
16. Dien, T.T., Sang, L.H., Thanh-Hai, N., Thai-Nghe, N.: Deep learning with data transformation and factor analysis for student performance prediction. *Int. J. Adv. Comput. Sci. Appl.* **11**(8), 711–721 (2020)
17. Al-Badarenah, A., Alsakran, J.: An automated recommender system for course selection. *Int. J. Adv. Comput. Sci. Appl.* **7**, 3 (2016)

18. Naren, J., Banu, M.Z., Lohavani, S.: Recommendation system for students' course selection. In: Soman, A.K., Shekhawat, R.S., Mundra, A., Srivastava, S., Verma, V.K. (eds.) Smart Systems and IoT: Innovations in Computing. SIST, vol. 141, pp. 825–834. Springer, Singapore (2020). https://doi.org/10.1007/978-981-13-8406-6_77
19. Bozyigit, A. et al.: Collaborative filtering based course recommender using OWA operators. In: 2018 International Symposium on Computers in Education (SIE) (2018)
20. Huang, L., et al.: A score prediction approach for optional course recommendation via cross-user-domain collaborative filtering. IEEE Access **7**, 19550–19563 (2019)
21. Thai-Nghe, N., Janecek, P., Haddawy, P.: A comparative analysis of techniques for predicting academic performance. In: 2007 37th Annual Frontiers in Education Conference - Global Engineering: Knowledge Without Borders, Opportunities Without Passports (2007)
22. Sang, L.H., Dien, T.T., Thai-Nghe, N., Thanh-Hai, N.: Predicting student's performance through deep learning using a multi-layer perceptron (in Vietnamese). Can Tho Univ. J. Sci. **56**(3), 20–28 (2020)
23. Nwankpa, C., Ijomah, W., Gachagan, A., Marshall, S.: Activation functions: comparison of trends in practice and research for deep learning. [ArXiv:1811.03378](https://arxiv.org/abs/1811.03378) (2018)
24. Nair, V., Hinton, G.E.: Rectified linear units improve restricted Boltzmann machines. In: ICML (2010)
25. Zeiler, M., et al.: On rectified linear units for speech processing. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 3517–3521 (2013)
26. Dahl, G., Sainath, T., Hinton, G.E.: Improving deep neural networks for LVCSR using rectified linear units and dropout. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 8609–8613 (2013)
27. Iqbal, Z., Qadir, J., Mian, A., Kamiran, F.: Machine learning based student grade prediction: a case study. [ArXiv:1708.08744](https://arxiv.org/abs/1708.08744) (2017)
28. Khanal, S.S., Prasad, P.W.C., Alsadoon, A., Maag, A.: A systematic review: machine learning based recommendation systems for e-learning. Educ. Inf. Technol. **25**(4), 2635–2664 (2019). <https://doi.org/10.1007/s10639-019-10063-9>
29. Thai-Nghe, N., Schmidt-Thieme, L.: Factorization forecasting approach for user modeling. Journal of Computer Science and Cybernetics. **31**(2), 133–148 (2015)
30. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer **42**, 30–37 (2009)
31. Koren, Y., Bell, R.: Advances in collaborative filtering. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 145–186. Springer, Boston, MA (2011). https://doi.org/10.1007/978-0-387-85820-3_5