

MINISTRY OF EDUCATION AND TRAINING
LAC HONG UNIVERSITY



HUYNH LY THANH NHAN

**AN APPROACH FOR ACADEMIC ADVISING USING
RECOMMENDER SYSTEM**

COMPUTER SCIENCE PHD THESIS ABSTRACT

Major: Computer Science

Code: 9480101

SCIENCE INSTRUCTOR

1. Assoc. Prof. Dr. NGUYEN THAI NGHE
2. Assoc. Prof. Dr. LE HUY THAP

Dong Nai – 2024

The work is completed at: Lac Hong University

Science instructor:

Assoc. Prof. Dr. Nguyen Thai Nghe

Assoc. Prof. Dr. Le Huy Thap

Review 1:

Review 2:

Review 3:

The thesis will be defended in front of the institutional examination council

.....

.....

At hour day month year.....

The thesis can be found at the library:

- The library of Lac Hong University

- National Library

TABLE OF CONTENTS

CHAPTER 1. INTRODUCTION	1
1.1 Overview of the thesis topic	1
1.2 Research problem and significance	1
1.2.1 The correlation between ranking prediction and students' academic performance prediction	1
1.2.2 The students' academic performance prediction problem.....	2
1.2.3 Recommendation for choosing elective courses.....	3
1.3 Research challenges.....	4
1.4 Research issues	5
1.5 Research objectives and approaches.....	5
1.6 Objects and scope of the study	6
1.7 Contributions of the thesis	6
1.8 Structure of the thesis	6
CHAPTER 2. THEORETICAL FOUNDATIONS AND RELATED WORKS	7
2.1 The Intelligent Tutoring System.....	7
2.2 The Recommender Systems	7
2.3 The Educational Data Mining.....	7
2.4 Related works	7
2.5 Chapter conclusion and discussion.....	8
CHAPTER 3. PREDICTING STUDENTS' ACADEMIC OUTCOMES USING A RECOMMENDER SYSTEM APPROACH	8
3.1 Approaching the problem of predicting students' academic performance using a recommender system.....	8
3.2 Collaborative filtering based on similar students (Student-kNNs)	8

3.3 Collaborative filtering based on similar courses (Course-kNNs)9

3.4 The Matrix Factorization method9

3.5 The Biased Matrix Factorization method10

3.6 Evaluation of results11

3.7 Chapter conclusion12

CHAPTER 4. DEEP MATRIX FACTORIZATION MODELS FOR
STUDENT PERFORMANCE PREDICTION 12

4.1 Problem formulation and solution methodology13

4.2 Integrating deep learning architecture into the matrix factorization.....13

4.3 Deep Biased Matrix Factorization method16

4.4 Evaluation of results17

4.5 Chapter conclusion17

CHAPTER 5. INTEGRATING DATA RELATIONSHIPS IN PREDICTING
STUDENT PERFORMANCE..... 18

5.1 Problem formulation and solution methodology18

5.2 Method for integrating the relationships of the student's classmates.....19

5.3 Method for integrating the relationships of courses20

5.4 Evaluation of results21

5.5 Chapter conclusion22

CHAPTER 6. CONCLUSION AND FUTURE WORKS 23

6.1 Conclusion23

6.2 Limitation and future works24

CHAPTER 1. INTRODUCTION

1.1 Overview of the thesis topic

Recently, the number of students who are forced to drop out of universities has been increasing at many institutions, particularly among third- and fourth-year students. One contributing factor is that these students often lack appropriate study plans. Therefore, identifying the potential of each student early on to help them create suitable study plans is one of the necessary and supportive research areas for academic advisors [1].

To assess the academic ability of students and provide reasonable course selection recommendations, many studies have been published on predicting the academic ability of students using popular technical methods such as Bayesian networks and decision trees [2][3], association rules and sequence-based association rules [4][5], genetic algorithms, game theory [6], or using rough set theory [7], as well as current trends such as deep learning models [8][9], or using situation-based reasoning (Case-based reasoning - CBR) [10].

1.2 Research problem and significance

1.2.1 The correlation between ranking prediction and students' academic performance prediction

Recently, the application of Recommender Systems (RS) in predicting students' academic performance (PSP) has been invested in research and development due to the similarity between predicting students' academic performance and ranking prediction in recommendation systems [11]. For example, students' academic performance in courses is measured by their scores, and product ratings by users are typically on a scale of 1 to 5 stars. Figure 1-1 illustrates the similarity between PSP and product rating.

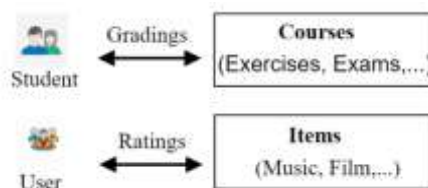


Figure 1-1: The correlation between ranking prediction and PSP

In a recommendation system, there are a list of users (u), a list of items (i), and ratings (r), which are users' evaluation scores on the items. Similarly, in the problem of predicting students' grades, there are lists of students (s), courses (c), and grades (g). Therefore, predicting user ratings in the rating prediction problem of RS is equivalent to predicting students' grades (Figure 1-2). This mapping is represented as: {User $\rightarrow$ Student}, {Item $\rightarrow$ Course}, {Ratings $\rightarrow$ Grading}.

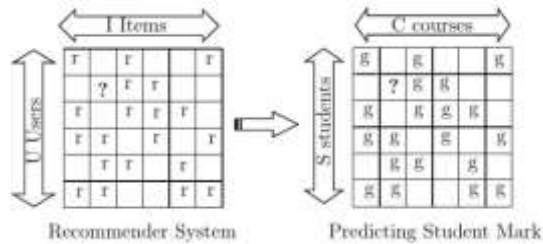


Figure 1-2: Mapping student model into recommender systems

1.2.2 The students' academic performance prediction problem

The model for predicting students' learning ability is personalized based on the idea of Collaborative Filtering (CF) methods in recommendation systems. Below are illustrations of the problem of predicting students' ability in three groups: Student-based filtering and Course-based filtering (Figure 1-3), and Model-based filtering (Figure 1-4). Student-based filtering relies on the similarity of abilities between two students and the results of a student who has studied to predict the results for a student who has not studied. Similarly, Course-based filtering will rely on similar courses to make recommendations for students who have studied a similar course.

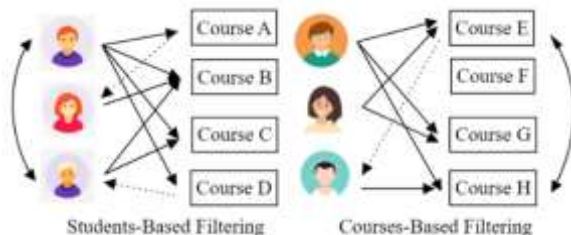


Figure 1-3: Collaborative filtering methods for the similarity

Model-based filtering uses latent factor models to predict the results of specific courses for each student.

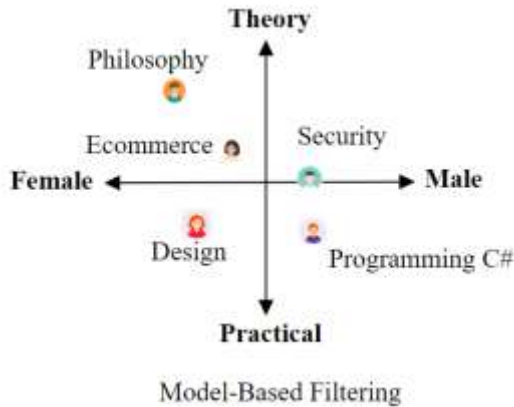


Figure 1-4: Model-based collaborative filtering method

1.2.3 Recommendation for choosing elective courses

Before students plan their study schedule for each semester, we explore their academic data to provide predictive results. These results serve as the basis for a system that recommends suitable elective courses while ensuring compliance with the program's curriculum constraints. For example, suppose we have five students: s_1 , s_2 , s_3 , s_4 , and s_5 , studying courses C_1 , C_2 , ..., C_n , $C-e_1$, $C-e_2$, and $C-e_3$, presented in a matrix like Figure 1-5, where each cell in the matrix contains the grade of the student for the corresponding course. Students who have not studied a course yet will fill the cell with a question mark "?". Among these courses, there are three elective courses: $C-e_1$, $C-e_2$, and $C-e_3$. Students need to choose two out of these three elective courses to obtain the most suitable results. Therefore, the system needs to recommend to student s_5 which two elective courses to study among $C-e_1$, $C-e_2$, and $C-e_3$.

After running the prediction algorithm for all students and all courses that they have not studied, we fill the matrix with the predicted grades. From the constraints regarding the number of credits or the number of elective courses that students need to study in a semester, the system will provide suitable recommendations.

		C - Courses						
		C1	C2		Cn	C-e1	C-e2	C-e3
S - Students	S1	2	2		2	1	4	?
	S2	3	2		4	?	2	2
	S3	1	2		?	3	?	1
	S4	2	3		2	3	?	2
	S5	1	4		4	?	?	?

Figure 1-5: Three courses that need to be predicted for the student s5

Using the example above, the system recommends that student s5 should study two elective courses: C-e1 and C-e2, because the predicted grades of two courses are higher than the grade of course C-e3 (3 and 4 > 2) as shown in Figure 1-6.

		C - Courses						
		C1	C2		Cn	C-e1	C-e2	C-e3
S - Students	S1	2	2		2	1	4	2
	S2	3	2		4	3	2	2
	S3	1	2		2	3	3	1
	S4	2	3		2	3	3	2
	S5	1	4		4	3	4	2

Figure 1-6: The predicted grades and recommendation method

1.3 Research challenges

Personalized Prediction: Each learner has different abilities and circumstances, so we need to provide personalized predictions for each student, while most current research only focuses on general patterns.

Biased Ratings: In the context of RS, many ratings are influenced by users or items, and this bias exists independently of the subjective relationship between users and items. This bias is manifested in systematic tendencies in the data, where some users will give higher (or lower) ratings than others or

some items are more (or less) accepted than others. Similarly, in the educational context, there are inconsistent ratings among different instructors (some instructors tend to grade higher or lower), as well as different levels of difficulty in different courses. This will affect the prediction results and the effectiveness of the prediction model, so it needs to be addressed.

Neglecting the relationships between users and between courses. Students in the same environment will influence each other, as well as courses with related content, which is a favorable point for us to exploit.

The accuracy of predictions can also be improved through integrated models. As science continues to develop, people always find new successful points in this field, which can be applied successfully in other fields.

1.4 Research issues

From analyzing the challenges and unresolved issues of related studies, the dissertation identifies several research issues, including: predicting students' learning abilities through personalization, addressing inconsistencies in evaluating students' learning abilities, exploring the potential of considering students' friendships in enhancing predictive accuracy, investigating the interrelationships among academic subjects, and improving prediction accuracy by integrating advanced models.

1.5 Research objectives and approaches

The main objective of this thesis is to propose new approaches to "student performance prediction" in order to develop a recommendation system for suitable course selection for students.

Objective 1: Propose data mining techniques for predicting students' learning ability to provide personalized course recommendations.

Objective 2: Propose methods to address the problem of uneven evaluation (as outlined in research challenge two).

Objective 3: Propose methods to integrate the relationships between students and the correlations between courses into the prediction model to improve the accuracy of predicting students' academic performance.

Objective 4: Propose methods to integrate the relationships between courses into the prediction model to improve the accuracy of predicting students' academic performance.

Objective 5: Propose the application of deep learning techniques to the prediction model to improve its effectiveness.

1.6 Objects and scope of the study

Study objects are prediction models and data mining. The scope of the thesis is to focus on methods for constructing prediction models, with experiments conducted on two datasets (Vietnam and international sources).

1.7 Contributions of the thesis

The first contribution of the thesis proposes a course recommendation model for students using a recommendation system approach, while also addressing the issue of inconsistent evaluation data (published in [CT1]).

The second contribution of the thesis proposes an integrated model that combines deep learning architecture with the matrix factorization method to improve prediction performance (published in [CT4][CT5]).

The third contribution of the thesis proposes a method that integrates user relationships into the student grade prediction model to leverage the influence of peer relationships on academic performance and improve prediction accuracy (published in [CT2]).

The fourth contribution of the thesis proposes methods that incorporate additional data, such as the correlation between courses, to improve the recommendation performance (published in [CT3]).

After the research and completion of the thesis, the author has published 02 papers in international conferences proceedings (IEEE/Scopus indexed), 01 paper in international conference proceedings (Springer/Scopus indexed), 01 paper in a journal (Scopus indexed 2020), 01 paper in an international journal.

1.8 Structure of the thesis

The thesis is structured as follows: chapter 1 - introduction, chapter 2 - overview of research status and related works, chapters 3, 4 & 5 - main research problems, chapter 6 - conclusion, list of published paper.

CHAPTER 2. THEORETICAL FOUNDATIONS AND RELATED WORKS

2.1 The Intelligent Tutoring System

The Intelligent Tutoring System (ITS) is a system capable of customizing and providing real-time feedback to learners without the intervention or assistance of a teacher [12]. ITS is composed of three different fields: computer science, education, and psychology.

2.2 The Recommender Systems

Recommender Systems (RS) is a type of information filtering system used to predict the preferences or ratings that users may give to an item they have not previously considered (an item could be a song, movie, book, product, article, news, etc.). RS is a subset of information filtering systems.

2.3 The Educational Data Mining

Accurate prediction of students' academic performance is highly useful in many different contexts in universities. For instance, academic advisors can identify outstanding candidates for talent contests or offer scholarships to encourage them to work harder. They can also identify students with poor performance (at risk of dropping out) and implement appropriate measures to support their learning or suggest elective courses in credit-based systems.

2.4 Related works

Due to the urgent nature of evaluating students' learning capabilities to provide reasonable counseling suggestions, many studies have been published on predicting students' academic abilities using traditional techniques such as decision trees, linear regression, support vector machine (SVM), Bayesian network models [13], sequential combination rules [4], genetic algorithms [6], or using rough set theory [7], applying RS to educational data mining and many other studies on EDM [14][15][16][17]. However, the RMSE error measurement results are still high, and there is a need to improve the accuracy of the algorithm.

2.5 Chapter conclusion and discussion

Based on the theoretical foundations and results of related studies, the dissertation provides an overview of the problem of recommendation solutions in student learning advising.

CHAPTER 3. PREDICTING STUDENTS' ACADEMIC OUTCOMES USING A RECOMMENDER SYSTEM APPROACH

The content of this chapter proposes predictive models for student learning outcomes based on a recommendation system approach, such as filtering tasks by similar students, similar courses, and in particular, two Matrix Factorization techniques and Biased Matrix Factorization. The experimental results demonstrate that the proposed methods have better accuracy than baseline methods. The content is synthesized from the publication [CT1].

3.1 Approaching the problem of predicting students' academic performance using a recommender system

Other problems in the field of data mining, such as building predictive systems for student learning outcomes, also follow the standard CRISP-DM (Cross Industry Standard Process for Data Mining) process. This process is designed to solve data mining problems and consists of six stages, similar to the waterfall model in analysis and design of information systems, including: understanding the problem, understanding the data, preparing the data, modeling, evaluating the model, and deploying the application.

3.2 Collaborative filtering based on similar students (Student-kNNs)

The user-based collaborative filtering method [18] is built on the idea of using the entire rating matrix to select a set of users with the most similar preferences to the current user. By combining ratings from similar users, it predicts ratings for the current user on items that have not been rated. Similarly, in the field of education, we assume that similar students will have similar abilities in similar subjects. Indeed, the user-based collaborative filtering method applied to items can be considered for predicting students' learning

abilities. The collaborative filtering method using k-nearest neighbors in the context of education is referred to as Student k-NNs.

3.3 Collaborative filtering based on similar courses (Course-kNNs)

The collaborative filtering method based on similar courses is different from the collaborative filtering method based on similar students. Instead of computing the similarity between students in the system and the current student s , this method calculates the similarity between the course to be predicted and the courses that student s has taken. To calculate the similarity between two courses, the method considers the set of students who have taken both courses. Then, the method selects a set of neighboring courses with the course for which the student s needs a score prediction. Finally, the method combines the scores of student s with this set of neighboring courses to provide a score prediction for the student s with the course to be predicted.

3.4 The Matrix Factorization method

Matrix factorization technique involves approximating a large matrix R into two matrices, W and H , which are much smaller in size than R , so that R can be reconstructed from these smaller matrices as accurately as possible [19], as illustrated in Figure 3-1, which means $R \approx WH^T$.

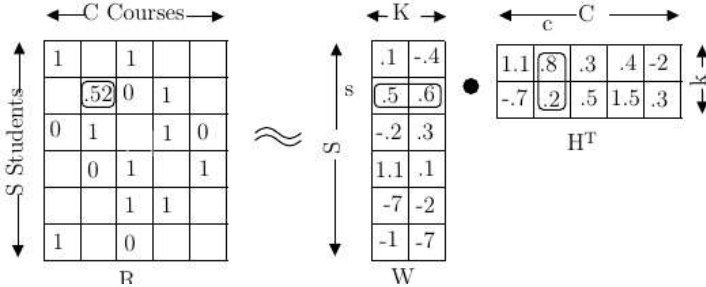


Figure 3-1: Matrix Factorization Model for Student Performance Prediction

$W \in \mathbb{R}^{|S| \times |K|}$ is a matrix where each row s is a vector containing k latent factors describing the student s , and $H \in \mathbb{R}^{|C| \times |K|}$ is a matrix where each row c is a vector containing k latent factors describing the course c

Let w_{sk} and h_{ck} are the elements, w_s and h_c are the vectors of W and H , respectively, then the grade g given by a student s to a course c is predicted:

$$\widehat{g}_{sc} = \sum_{k=1}^K w_{sk} h_{ck} = w_s h_c^T \quad (3.1)$$

The main issue of this technique is how to find optimal values for the parameters W and H given a criterion such as Root Mean Squared Error (RMSE), which is determined by

$$RMSE = \sqrt{\frac{1}{|D^{test}|_{s,c,g \in D^{test}}} \sum (g_{sc} - \widehat{g}_{sc})^2} \quad (3.2)$$

Optimizing the objective function using Stochastic Gradient Descent (SGD) method.

$$o^{MF} = \sum_{(s,c,g) \in D^{train}} (g_{sc} - \sum_{k=1}^K \widehat{w}_{sk} \widehat{h}_{ck})^2 + \lambda (\|W\|_F^2 + \|H\|_F^2) \quad (3.3)$$

Where $\|\cdot\|_F^2$ is a Frobenius norm, λ is a regularization weight ($0 \leq \lambda < 1$). The term $\lambda \cdot (\|W\|_F^2 + \|H\|_F^2)$ is used for preventing over-fitting. With this new error function, the values of w_{sk} and h_{ck} are updated respectively

$$w' = w_{sk} + \beta (2e_{sc} h_{ck} - \lambda w_{sk}) \quad (3.4)$$

$$h' = h_{ck} + \beta (2e_{sc} w_{sk} - \lambda h_{ck}) \quad (3.5)$$

Where $e_{sc} = g_{sc} - \widehat{g}_{sc}$ is the prediction error, β is the learning rate.

After the training process, two optimized matrices W and H are obtained, and the prediction process is carried out. The prediction process is calculated as follows:

$$\widehat{g}_{sc} = \sum_{k=1}^K w_{sk} h_{ck} = w_s h_c^T \quad (3.6)$$

3.5 The Biased Matrix Factorization method

To address the issue of objective evaluation and reduce the gap between different high or low requirements of different courses, as well as minimize the biased suggestions based on the strengths or weaknesses of students in certain subjects, we used the Matrix Factorization algorithm combined with a bias offset metric to obtain the Biased Matrix Factorization (BMF) algorithm, a method published in [CT1]. To predict the grade of student s for course c , it is represented by the following formula.

$$\widehat{g}_{sc} = \mu + b_s + b_c + \sum_{k=1}^K w_{sk} h_{ck} \quad (3.7)$$

With the value of μ as the global mean value, which is the average gradings of all students across all courses in the training dataset.

$$\mu = \frac{\sum_{(s,c,g) \in D^{train}}}{|D^{train}|} \quad (3.8)$$

The student effect (student bias) models how good/clever/bad a student is (i.e., how likely is the student to perform a course correctly). The value of b_s is the average deviation of a student's ability from the global mean value.

$$b_s = \frac{\sum_{(s',c,g) \in D^{train} | s'=s} (g-\mu)}{|\{(s',c,g) \in D^{train} | s'=s\}|} \quad (3.9)$$

The course effect (course bias) models how difficult/easy the course is (i.e., how likely is the course to be performed correctly). The value of b_c is the average deviation of a course's requirement from the global mean value

$$b_c = \frac{\sum_{(s,c',g) \in D^{train} | c'=c} (g-\mu)}{|\{(s,c',g) \in D^{train} | c'=c\}|} \quad (3.10)$$

As there are changes in the bias values, the objective function also changes according to the following formula:

$$o^{BMF} = \sum_{(s,c,g) \in D^{train}} (g_{sc} - \mu - b_s - b_c - \sum_{k=1}^K w_{sk} h_{ck})^2 + \lambda (\|W\|_F^2 + \|H\|_F^2 + b_s^2 + b_c^2) \quad (3.11)$$

After the training process, we obtain two optimized matrices W and H , and then the prediction process is performed.

3.6 Evaluation of results

Dataset - The ASSISTments dataset is published by the ASSISTments platform. After preprocessing, this dataset contains 8519 students (users), 35978 tasks (item), and 1011079 scores (ratings).

Evaluation - To evaluate the algorithm, we use the hold-out evaluation method: randomly selecting 2/3 of the dataset for training and the remaining 1/3 for testing. Since the task is to predict the learning outcomes of students in the form of rating prediction, the most suitable evaluation metric is Root Mean Squared Error (RMSE). Moreover, major awards in the RS field use RMSE for evaluation, such as Netflix Prize, KDD Cup 2010, etc.

Hyper-parameter Search - The accuracy of prediction depends on the parameters provided to the algorithm. If the parameters are not appropriate, the

accuracy of the prediction will not be good even if the algorithm is correct. Therefore, finding the best parameters is very important. Hyper-parameter search consists of two stages: Coarse Search and Granularity Search.

We conducted four experiments, and the experimental results are displayed in Figure 3-2. Comparing the RMSE error of different methods, the graph shows that the error of the proposed approach is the smallest on the CTU dataset. The smallest error indicates the best model.

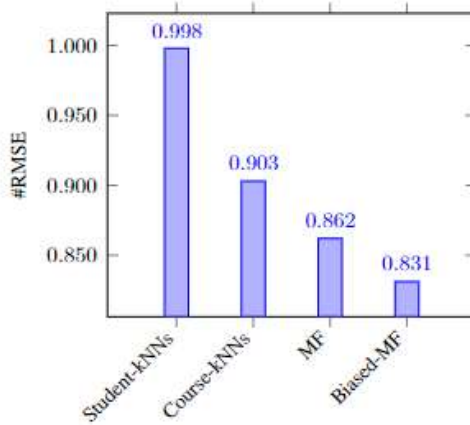


Figure 3-2: Compare the RMSE error of the methods

3.7 Chapter conclusion

This chapter has introduced an approach using the MF techniques to predict students' learning outcomes. Testing on standard datasets showed that the proposed process worked well. The application of the method yielded promising results. However, the prediction accuracy could be further improved by the enhanced methods presented in the next Chapter 4.

CHAPTER 4. DEEP MATRIX FACTORIZATION MODELS FOR STUDENT PERFORMANCE PREDICTION

The content of this chapter proposes the integration of deep learning models into the matrix factorization technique to improve the accuracy of predictions, which is highly suitable for developing applications in the

educational recommendation system of the learning advisor. The content of this chapter is synthesized from the publication [CT4]

4.1 Problem formulation and solution methodology

Recently, there has been increasing interest in applying deep learning (DL) to educational data mining. The combination of DL with Matrix Factorization (MF) technique holds great promise for achieving significant results. In addition, as educational data is being collected more than ever, there is a potential for applying big data processing techniques.

4.2 Integrating deep learning architecture into the matrix factorization

The main idea of integrating deep learning architecture into the MF technique is to reconstruct the matrix factorization process, iteratively approximating the output matrix until the best result is achieved. Each step in the process is a basic method (MF); the complete model is the sum of matrices stored on the stack of factorization steps.

Figure 4-1 illustrates the operation of the Deep Matrix Factorization method (DMF). As in the classical MF method, this matrix X will also be called $X = G^0$ which is the beginning of the process. Approximating a matrix $X \in G^{|S| \times |C|}$ by a product of two smaller matrices, W^0 and H^0 , is a form $\hat{G}^0 = W^0 \cdot H^0$. The matrix $\hat{G}^0$ provides all the predicted gradings, and they were stored in the stack at the first step. At this step, the recursive is begins. A new matrix $G^1 = X - \hat{G}^0$ is built by computing the attained errors between the original gradings matrix X and the predicted gradings stored in $\hat{G}^0$. A factorization again approximates this new matrix G^1 into two new small rank matrices $\hat{G}^1 = W^1 \cdot H^1$, which produces the errors at the second step $G^2 = G^1 - \hat{G}^1$. This process is repeated many times by generating and factorizing successive error matrices $G^1, \dots, G^T$. Presumably, this sequence of error matrices converges to zero, so we get preciser predictions as we add new layers to the model.

In a similar manner to the standard MF method (section 3.4), this method can be formalized and the approximation can be expressed as follows:

$$G^0 \approx \hat{G}^0 = W^0 \cdot H^0 \quad (4.1)$$

To implement the DMF model, we subtract the approximation performed by $\hat{G}^0$ to the original matrix X , to obtain a new sparse matrix G^1 that contains the prediction error at the first iteration:

$$G^1 = R - \hat{G}^0 = G^0 - W^0 \cdot H^0 \quad (4.2)$$

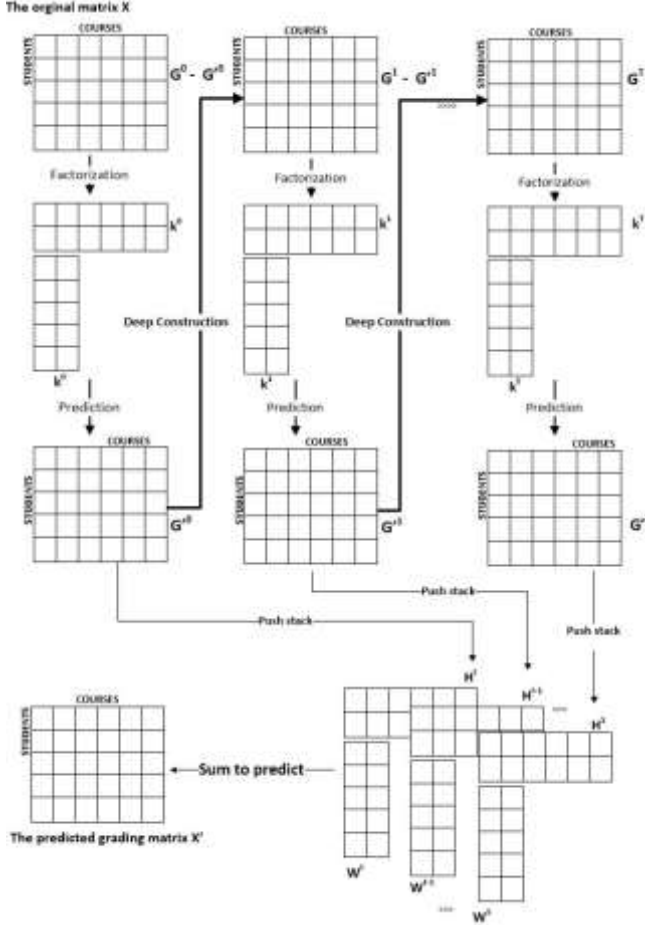


Figure 4-1: Deep Matrix Factorization for student performance prediction

Note that positive values in the matrix G^1 mean that the prediction is low and need to be increased. Similarly, the negative values in the matrix G^1 mean that the prediction is high and need to be decreased. Indeed, this adjustment is the main idea of applying the deep learning approach. To do this, we need to perform a new factorization to the error matrix G^1 in such a way:

$$G^1 \approx \hat{G}^1 = W^1 \cdot H^1 \quad (4.3)$$

The approximation process in each step is performed similarly. However, note that $k^1 \neq k^0$ to get various resolutions in the factorization. In the general case, if we computed at $t - 1$ steps of the procedure, the t^{th} matrix of errors can be determined:

$$G^t = G^{t-1} - \hat{G}^{t-1} = G^{t-1} - W^{t-1} \cdot H^{t-1} \quad (4.4)$$

At the step t , we are also factorizing into matrices W^t and H^t have the k^t latent factors until the sequence of error matrices converge to zero.

$$G^t \approx \hat{G}^t = W^t \cdot H^t \quad (4.5)$$

Once the deep factorization process ends after T steps, the original grading matrix X can be reconstructed by adding the estimates of the errors as:

$$R \approx \hat{R} = \hat{G}^0 + G^1 = \hat{G}^0 + \hat{G}^1 + G^2 = \dots = \hat{G}^0 + \hat{G}^1 + \hat{G}^2 + \dots + \hat{G}^T = W^0 \cdot H^0 + W^1 \cdot H^1 + \dots + W^T \cdot H^T = \sum_{t=0}^T W^t \cdot H^t \quad (4.7)$$

The error function for the DMF now becomes:

$$O^{DMF} = \sum_{t=0}^T \left(\sum_{(s,c,g) \in D^{train}} (g_{sc}^t - \sum_{k=1}^K w_{sk}^t h_{kc}^t)^2 + \lambda^t (\|W^t\|_F^2 + \|H^t\|_F^2) \right) \quad (4.8)$$

Where λ^t is the regularization parameter of the step t to avoid overfitting.

The error function O^{DMF} can be derived to w_s^t and h_c^t resulting in the following updated rules for training the model parameters. The w_{sk}^t and h_{ck}^t are updated by the equations below (where $e_{sc}^t = g_{sc}^t - \hat{g}_{sc}^t$, and $w_{sk}^{t'}$ is the updated value of w_{sk}^t , and $h_{ck}^{t'}$ is the updated value of h_{ck}^t). With the new error function of the DMF, The values of $w_{sk}^{t'}$ and $h_{ck}^{t'}$ are updated respectively

$$h_{ck}^{t'} = h_{ck}^t + \beta(2e_{sc}w_{sk}^t - \lambda h_{ck}^t) \quad (4.9)$$

$$w_{sk}^{t'} = w_{sk}^t + \beta(2e_{sc}h_{ck}^t - \lambda w_{sk}^t) \quad (4.10)$$

Where β^t is the learning rate hyper-parameter of step t to control the learning speed. In this way, after finishing the nested factorization, all the predicted ratings are collected in the matrix $\hat{X} = (\hat{g}_{sc})$, where the predicted the grading of the student s to the course c is given by:

$$\hat{g}_{sc} = \sum_{t=0}^T \sum_{k=1}^K w_{sk}^t h_{kc}^t = \sum_t w_s^t * (h_c^t)^T \quad (4.11)$$

Proposed Algorithm

Algorithm 4.1: Predicting Student Performance DMF

Input: D^{train} , K, betas, lamdas, stopping condition, depth

Output: W, H

1. Let $s \in S$ be a student, $c \in C$ a course, $g \in G$ a grade
 2. Let $W[|S|][K]$, $H[|C|][K]$ be latent factors of students, courses
 3. $W \leftarrow N(0, \sigma^2)$ and $H \leftarrow N(0, \sigma^2)$
 4. $k, t, \beta, \lambda \leftarrow \text{pop}(K, T, \text{Betas}, \text{Lamdas})$
 5. while (Stopping criterion is NOT met) do
 6. for each (s, c, g_{sc}) from D^{train}
 7. $\widehat{g}_{sc} \leftarrow \sum_k^K (W[s][k] * H[c][k])$
 8. $e_{sc} = g_{sc} - \widehat{g}_{sc}$
 9. for $k = 1..K$ do
 10. $W[s][k] = W[s][k] + \beta * (2e_{si} * H[c][k] - \lambda * W[s][k])$
 11. $H[c][k] \leftarrow H[c][k] + \beta * (2e_{sc} * W[s][k] - \lambda * H[c][k])$
 12. end for
 13. end for
 14. end while
 15. if (is-empty(K)) return new stack ($\langle W, H \rangle$)
 16. else
 17. $\hat{G} = G - W \cdot H$
 18. Params-Stack=Deep- Matrix-Factoriztion ($\hat{G}, K, T, \text{Betas}, \text{Lamdas}$)
 19. Return push($\langle W, H \rangle$, Params-Stack)
 20. end else.
 21. end function.
-

4.3 Deep Biased Matrix Factorization method

Similar to the DMF method (presented in section 4.2), this proposed method uses the Biased-MF as input for the multi-layer decomposition process instead of using MF. This method is called Deep Biased Matrix Factorization (DBMF). This method is to synthesize research results from the work [CT5].

4.4 Evaluation of results

The hyperparameters for the models on the ASSISTments dataset and the complexity of these algorithms are presented in Table 4.1.

Table 4.1: Hyper-parameters on ASSISTments dataset

Methods	Hyper Parameter	Complexity
MF	$\beta=0.03$, #iter=50, $k=4$, $\lambda=0.05$	$O(s \times c \times i)$
DMF	Deep (d)= 4; numFactors (k)= [3, 6, 3, 3]; numIters (i)=[50, 50, 50, 50]; learningRate (β)=[0.01, 0.1, 0.1, 0.1]; regularization (λ) = [0.01, 0.1, 0.1, 0.01]	$O(s \times c \times i \times d)$
DBMF	DMF, RegS, RegC	$O(s \times c \times i \times d)$

The experimental results are shown in (Figure 4-2), which indicates that the RMSE of the proposed approach (DBMF) is the smallest (0.332) on the Assistment dataset. The smaller the error, the better the model performance.

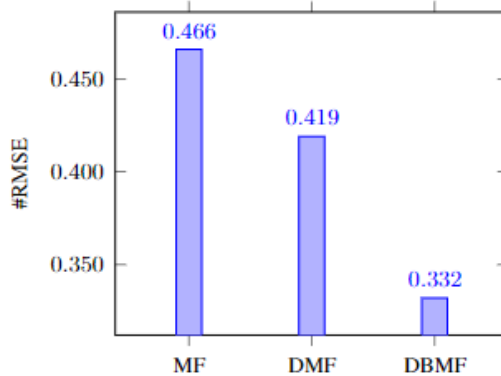


Figure 4-2: Experimental results on the dataset ASSISTment

4.5 Chapter conclusion

This chapter has introduced an approach using deep learning architecture to implement the MF and Biased-MF methods in order to improve the accuracy of predictions. Applying deep learning architecture for the MF technique may slow down the training time, but parallel algorithms can be applied to solve this problem. Integrating additional information (meta-data) to improve prediction accuracy is presented in the next Chapter 5.

CHAPTER 5. INTEGRATING DATA RELATIONSHIPS IN PREDICTING STUDENT PERFORMANCE

This chapter presents in detail the methods to improve the accuracy of predictions, such as integrating the relationships of students (based on their relationships with classmates) and the correlations of courses (based on the knowledge and skills of each course). The results are tested on published and internal datasets, all showing better results compared to non-integrated methods [CT2][CT3].

5.1 Problem formulation and solution methodology

Integrating the relationships of the student's classmates

With the increasing development of online social networks, integrating social networks into RS has become a topic of interest for many researchers [20]. The use of social network integration is different from finding similar users, as it utilizes the relationship components of another system such as Facebook, Twitter, Zalo, etc. to supplement information and improve prediction accuracy. Assuming that 'If A is a friend of B, and A performs well in studying and is diligent, it may influence B and vice versa'. To test this assumption, the thesis integrated the friendships of students into the model predicting their academic performance, and achieved better results

Integrating course relationships

Similarly, adding relationships between items to the prediction model in a recommender system also improves prediction performance [21]. With a similar idea: 'If two problems A and B require similar skills, and a student can solve problem A, then they may be able to solve problem B'. Therefore, effectively exploiting these relationships can enhance the prediction accuracy of a student's academic performance.

The advantage of having data suitable for integration

In the ASSISTments dataset, the Student_Class_Id field (the student's class) is used as a measure of students' friendships, and the Skill_Id field (the

required skill for the course) is used to find relationships between course. These two fields are significant in improving the effectiveness of prediction.

5.2 Method for integrating the relationships of the student's classmates

To integrate the relationships of students into matrix factorization technique, the friendships among students in the same class will be transformed into a binary relationship matrix (Figure 5-1).

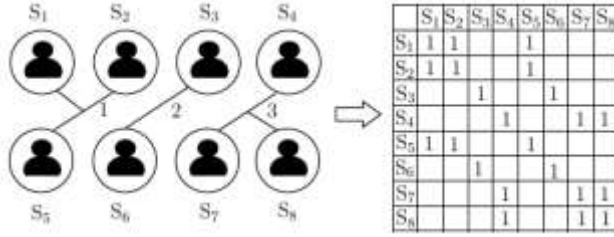


Figure 5-1: The transformation of the friendship into relationship matrix

In the relationship-based approach, each student s has a set N_s representing their friend relationships with s . The value $T_{s,n}$ represents the probability of the relationship between student s and student n , and is a binary number; $T = 0$ means that the two individuals s and n are in different classes, and $T = 1$ means that they are in the same class. Figure 5-2 illustrates the model of integrating user relationships into the MF technique.

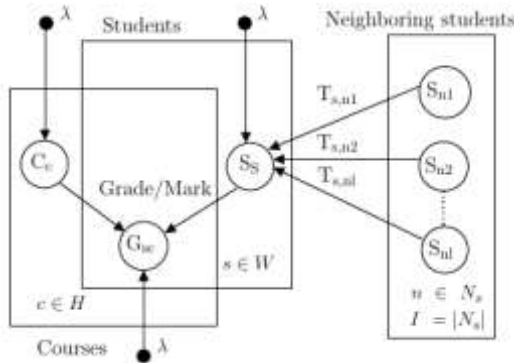


Figure 5-2: Model for integrating user relationships into MF techniques

To incorporate the relationship, we only need to replace the matrix W_s with $\widehat{W}_s$ using (5.1) and then proceed similarly to the basic MF technique.

$$\hat{w}_s = \frac{\sum_{n \in N_s} T_{s,n} w_n}{\sum_{n \in N_s} T_{s,n}} = \frac{1}{|N_s|} \sum_{n \in N_s} w_n \quad (5.1)$$

Where $\hat{W}_s$ is the estimated latent feature vector of s given the feature vectors of his direct neighbors. Since $T_{s,n}$ is a binary matrix so that all non-zero values of $T_{s,n} = 1$ and $\sum_{n \in N_s} T_{s,n} = |N_s|$. To integrate relationship into the MF, we need replace W_s by $\hat{W}_s$ in the prediction function, and it become:

$$\hat{g}_{sc} = \sum_{k=1}^K \hat{w}_{sk} h_{ck} = \hat{w}_s \cdot h_c^T \quad (5.2)$$

The objective function of the SocialMF technique is still calculated as in the MF technique, but with an additional coefficient for the relationship:

$$\begin{aligned} O^{\text{SocialMF}} = \sum_{(s,c,g) \in D^{\text{train}}} (g_{sc} - \sum_{k=1}^K \hat{w}_{sk} h_{ck})^2 + \lambda (\|W\|_F^2 + \|H\|_F^2) + \\ \lambda_T \sum_{s=1}^S \left(w_s - \frac{1}{|N_s|} \sum_{n \in N_s} w_n \right)^2 \end{aligned} \quad (5.3)$$

The values of w and h are also updated after each iteration as follows:

$$\begin{aligned} w'_{sk} = w_{sk} + \beta (2e_{sc} h_{ck} - \lambda w_{sk}) + \lambda_T \left(w_{sk} - \frac{1}{|N_s|} \sum_{n \in N_s} w_{nk} \right) - \\ \lambda_T \left(\sum_{t \in S_s} \frac{1}{|N_s|} \left(w_{tk} - \frac{1}{|N_s|} \sum_{w \in N_s} w_{wk} \right) \right) \end{aligned} \quad (5.4)$$

$$h'_{ck} = h_{ck} + \beta (2e_{sc} w_{sk} - \lambda h_{ck}) \quad (5.5)$$

Where $e_{sc} = g_{sc} - \hat{g}_{sc}$; β is the learning rate; λ is the regularization for matrices W and H ; λ_T is the regularization for the relationship matrix T ; S_s is the set of all students; N_s is the set of classmates of student s ; $|N_s|$ is the total number of classmates with student s . After the optimization process, we obtain the optimized values of matrices W and H . The prediction function is as:

$$\hat{g}_{sc} = \sum_{k=1}^K w_{sk} h_{ck} = w_s \cdot h_c^T \quad (5.6)$$

5.3 Method for integrating the relationships of courses

Similarly, we need to convert this relationship into a binary relationship matrix R . Each course has a set R_c representing the relationship between course c and course m . The value of R_{cm} is either 0 or 1. If $R_{cm} = 1$, then courses c and m are related, otherwise $R_{cm} = 0$. The MF model still serves as the basis for integrating the relationship between courses into the model.

Similarly to SocialMF, we calculate the value of $\hat{h}$ to replace h as:

$$\hat{h}_c = \frac{\sum_{m \in R_c} R_{cm} h_m}{\sum_{m \in R_c} R_{cm}} = \frac{1}{|R_c|} \sum_{m \in R_c} h_m \quad (5.7)$$

The prediction still follows the MF formula, but replaces h with $\hat{h}$

$$\hat{g}_{sc} = \sum_{k=1}^K w_{sk} \hat{h}_{ck} = w_s \cdot \hat{h}_c^T \quad (5.8)$$

The new objective function (CRMF) is calculated:

$$O^{\text{CRMF}} = \sum_{(s,c,g) \in D^{\text{train}}} (g_{sc} - \sum_{k=1}^K \hat{w}_{sk} \hat{h}_{ck})^2 + \lambda (\|W\|_F^2 + \|H\|_F^2) + \lambda_R \sum_{c=1}^C \left(h_c - \frac{1}{|M_c|} \sum_{m \in M_c} h_m \right)^2 \quad (5.9)$$

The values of w and h are also updated after each iteration as follows:

$$w'_{sk} = w_{sk} + \beta (2e_{sc} h_{ck} - \lambda w_{sk}) \quad (5.10)$$

$$h'_{ck} = h_{ck} + \beta (2e_{sc} w_{sk} - \lambda h_{ck}) + \lambda_R \left(h_{ck} - \frac{1}{|M_c|} \sum_{m \in M_c} h_{mk} \right) - \lambda_R \left(\sum_{r \in C_c} \frac{1}{|M_c|} \left(h_{rk} - \frac{1}{|M_c|} \sum_{y \in R_c} h_{yk} \right) \right) \quad (5.11)$$

Where $e_{sc} = g_{sc} - \hat{g}_{sc}$, β is the learning rate, λ is the regularization for matrices W and H ; λ_R is the regularization for the relationship matrix R ; C_c is the set of all courses, M_c is the set of courses that are related to each other; $|M_c|$ is the total number of courses in the same related group. After the optimization process, the prediction function is as follows:

$$\hat{g}_{sc} = w_s \cdot \hat{h}_c^T \quad (5.12)$$

5.4 Evaluation of results

The ASSISTments dataset after processing includes 8519 students, 35978 problems, and 1011079 performances. The CTU University dataset includes 4017 students, 353 courses, and 279536 scores. We use the hold-out cross-validation method and the (RMSE) measure for experiments evaluation. The hyperparameters and complexity described in detail (Table 5.1).

Table 5.1: Hyper-parameters on ASSISTments dataset

Methods	Hyper-Parameters	Complexity
Student-kNNs	k=5, simMeasure=Cosine	$O(s^2 \times c)$
Course-kNNs	k=5, simMeasure=Cosine	$O(c^2 \times s)$
MF	$\beta=0.01$, $i=60$, $k=32$, $\lambda=0.1$	$O(s \times c \times i)$
Social-MF	$k=80$, $\lambda=3$, $\beta=0.01$, SocialReg=5, $i=500$	$O(s^3 \times c \times i)$
CRMF	$k=80$, $\lambda=3$, $\beta=0.01$, ItemReg=3, $i=300$	$O(c^3 \times s \times i)$

From the comparison chart (Figure 5-3), it can be seen that when applying the user relationship integration algorithm to the predicting students'

academic performance, it achieved the lowest RMSE error compared to other algorithms when experimented on 2 datasets from Vietnam and international.

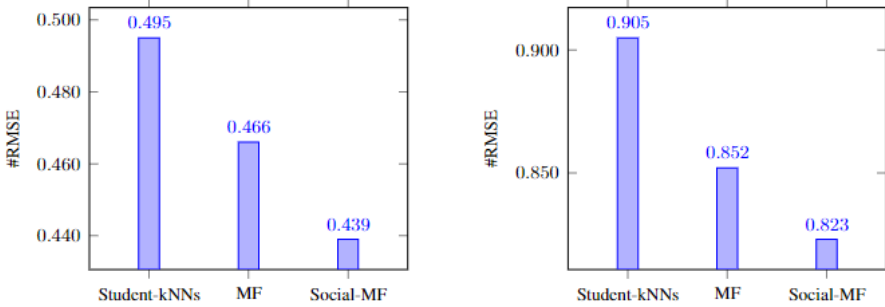


Figure 5-3: Comparison of RMSE error on 2 datasets CTU and ASSISTment

Similarly, from the chart (Figure 5-4), it can be seen that when applying the CRMF algorithm to the problem of predicting student performance, it achieves the lowest RMSE error (0.423) compared to other algorithms.

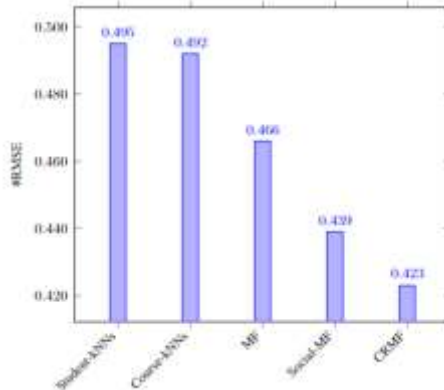


Figure 5-4: The RMSE error comparison chart of integrated algorithms

5.5 Chapter conclusion

The thesis has introduced two approaches to integrate the social relationships among students and the correlations among courses into predicting students' academic performance, illustrated through two techniques, SocialMF and CRMF. With these approaches, we can leverage additional information from the existing dataset to integrate into the prediction model, improving the prediction results.

CHAPTER 6. CONCLUSION AND FUTURE WORKS

6.1 Conclusion

The thesis aims to address a topic of theoretical and practical significance in computer science that is of particular interest to the research community and educators, which is the investigation of recommendation solutions in academic advising and the development of its applications.

Theoretical contributions

The thesis has presented fundamental and expanded research on educational data mining techniques in predicting students' academic performance personalized to their needs, and has shown that applying recommendation systems in academic advising has somewhat positively supported students in choosing suitable courses for each individual student.

Although the application of recommender systems in predicting student academic performance has shown promising results, the issue of improving the accuracy of algorithms and utilizing additional information remains open. Therefore, the thesis proposes some methods to integrate data relationships into matrix factorization techniques to improve the prediction effectiveness. Indeed, integrating interaction information (such as student friendship relationships, relationships between courses) into the prediction model increases the accuracy of the predictions. Through experiments, it has been shown that these integration methods are more effective than other methods.

Currently, deep learning is a popular technique in prediction, recognition, and classification problems applied in many fields. With the idea based on the superiority of deep learning techniques, we combine deep learning architecture with matrix factorization technique in the recommendation system to provide more accurate prediction results. From there, it can be applied to the problem of predicting students' academic performance to serve as a basis for recommending suitable elective courses.

Practical contributions

In practical terms, to apply data mining techniques to solve the problem of predicting students' learning abilities and supporting study advice

recommendations, it is necessary to build an intuitive tool (website) that assists students in making decisions about planning their studies based on their own abilities. Therefore, the thesis proposes a model architecture to build a flexible and easily upgradable or improved course selection recommendation system.

6.2 Limitation and future works

Limitations

Although the proposals put forward in the thesis have addressed the problem of suggesting courses for students, which is an important and time-consuming task for academic advisors, there are still some limitations that have not been resolved in the proposed solutions:

- The problem of suggesting courses to students in an order that is suitable for each individual student.
- The issue of limited supplementary data, such as students' social network data or digitized content data for courses.

Future works

The learning capability of students will change over time, affecting the prediction results. Therefore, researching the technique of time series matrix factorization (Time Series MF) will be a potential direction for the study.

Moreover, the order of courses that students take will also affect the learning outcomes and the prediction results of the algorithm. Predicting results using sequential matrix factorization technique (Sequence MF) is also a promising development direction.

In addition to applying matrix factorization techniques to predict student learning outcomes with great success, there are still open questions about applying modern methods to solve this problem, such as distance measurement in recommendation systems, or new methods such as Graph Neural Networks that leverage the prediction power of deep learning for rich data structures that describe objects and their relationships as points connected by edges in a graph. In the future, these are potential methods to solve the problem of predicting student learning outcomes.

REFERENCES

- [1] A. C. Rivera, M. Tapia-Leon, and S. Lujan-Mora, "Recommendation Systems in Education: A Systematic Mapping Study," *Advances in Intelligent Systems and Computing*, vol. 721, pp. 937–947, jan 2018. https://link.springer.com/chapter/10.1007/978-3-319-73450-7_89
- [2] J. S. Gil, A. J. P. Delima, and R. N. Vilchez, "Predicting students' dropout indicators in public school using data mining approaches," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 9, no. 1, pp. 774–778, 2020, doi: 10.30534/ijatcse/2020/110912020.
- [3] J. Kabathova and M. Drlik, "Towards Predicting Student's Dropout in University Courses Using Different Machine Learning Techniques," *Appl. Sci. 2021, Vol. 11, Page 3130*, vol. 11, no. 7, p. 3130, Apr. 2021, doi: 10.3390/APP11073130.
- [4] H. Q. Nguyen, T. T. Pham, V. Vo, B. Vo, and T. T. Quan, "The predictive modeling for learning student results based on sequential rules," *Int. J. Innov. Comput. Inf. Control*, vol. 14, no. 6, pp. 2129–2140, Dec. 2018, doi: 10.24507/IJICIC.14.06.2129.
- [5] H. H. Varzaneh, B. S. Neysiani, H. Ziafat, and N. Soltani, "Recommendation Systems Based on Association Rule Mining for a Target Object by Evolutionary Algorithms," *Emerg. Sci. J.*, vol. 2, no. 2, pp. 100–107, May 2018, doi: 10.28991/ESJ-2018-01133.
- [6] A. Esteban, A. Zafra, and C. Romero, "Helping university students to choose elective courses by using a hybrid multi-criteria recommendation system with genetic optimization," *Knowledge-Based Syst.*, vol. 194, p. 105385, Apr. 2020, doi: 10.1016/J.KNOSYS.2019.105385.
- [7] B. C. Madeira, T. Tasci, and N. Celebi, "Prediction of Student Performance Using Rough Set Theory And Backpropagation Neural Networks," *Eur. Sci. Journal, ESJ*, vol. 17, no. 7, pp. 1–1, Feb. 2021, doi: 10.19044/ESJ.2021.V17N7P1.
- [8] T. T. Dien, S. H. Luu, N. Thanh-Hai, and N. Thai-Nghe, "Deep Learning with Data Transformation and Factor Analysis for Student Performance Prediction," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 8, pp. 711–721, 2020, doi: 10.14569/IJACSA.2020.0110886.
- [9] X. Li, X. Zhu, X. Zhu, Y. Ji, and X. Tang, "Student Academic Performance Prediction Using Deep Multi-source Behavior Sequential Network," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12084 LNAI, pp. 567–579, May 2020, doi: 10.1007/978-3-030-47426-3_44.
- [10] M. Hasnawi, N. Kurniati, S. H. Mansyur, Irawati, and T. Hasanuddin, "Combination of Case Based Reasoning with Nearest Neighbor and Decision Tree for Early Warning System of Student Achievement," *Proc. - 2nd East Indones. Conf. Comput. Inf. Technol. Internet Things*

Ind. EIconCIT 2018, pp. 78–81, Nov. 2018, doi: 10.1109/EICONCIT.2018.8878512.

- [11] N. Thai-Nghe and L. Schmidt-Thieme, “Multi-relational Factorization Models for Student Modeling in Intelligent Tutoring Systems,” in *Proceedings - 2015 IEEE International Conference on Knowledge and Systems Engineering, KSE 2015*, Jan. 2015, pp. 61–66, doi: 10.1109/KSE.2015.9.
- [12] B. Park Woolf, L. New York, S. Diego, and M. Kaufmann, *Building Intelligent Interactive Tutors Student-centered strategies for revolutionizing e-learning*. 2009.
- [13] X. Du, J. Yang, J. L. Hung, and B. Shelton, “Educational data mining: a systematic review of research and emerging trends,” *Inf. Discov. Deliv.*, vol. 48, no. 4, pp. 225–236, Oct. 2020, doi: 10.1108/IDD-09-2019-0070/FULL/XML.
- [14] M. C. Mihaescu and P. S. Popescu, “Review on publicly available datasets for educational data mining,” *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, vol. 11, no. 3, p. e1403, May 2021, doi: 10.1002/WIDM.1403.
- [15] N. Thai-Nghe, L. Drumond, A. Krohn-Grimberghe, and L. Schmidt-Thieme, “Recommender system for predicting student performance,” *Procedia Comput. Sci.*, vol. 1, no. 2, pp. 2811–2819, 2010, doi: 10.1016/J.PROCS.2010.08.006.
- [16] R. Lara-Cabrera, Á. González-Prieto, and F. Ortega, “Deep Matrix Factorization Approach for Collaborative Filtering Recommender Systems,” *Appl. Sci. 2020, Vol. 10, Page 4926*, vol. 10, no. 14, p. 4926, Jul. 2020, doi: 10.3390/APP10144926.
- [17] T. T. Dien, N. Thanh-Hai, and N. T. Nghe, “Deep Matrix Factorization for Learning Resources Recommendation,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12876 LNAI, pp. 167–179, Sep. 2021, doi: 10.1007/978-3-030-88081-1_13.
- [18] X. Su and T. M. Khoshgoftaar, “A Survey of Collaborative Filtering Techniques,” *Advances in Artificial Intelligence*, vol. 2009, pp. 1–19.
- [19] Y. Koren, R. Bell, and C. Volinsky, “Matrix factorization techniques for recommender systems,” *Computer (Long. Beach. Calif.)*, vol. 42, no. 8, pp. 30–37, 2009, doi: 10.1109/MC.2009.263.
- [20] R. Chen *et al.*, “A Novel Social Recommendation Method Fusing User’s Social Status and Homophily Based on Matrix Factorization Techniques,” *IEEE Access*, vol. 7, pp. 18783–18798, 2019, doi: 10.1109/ACCESS.2019.2893024.
- [21] Y. Yu, C. Wang, H. Wang, and Y. Gao, “Attributes coupling based matrix factorization for item recommendation,” *Appl. Intell.*, vol. 46, no. 3, pp. 521–533, Apr. 2017, doi: 10.1007/S10489-016-0841-8.

PUBLISHED WORKS

[CT1] Huynh-Ly Thanh-Nhan, Huu-Hoa Nguyen, and Nguyen Thai-Nghe. (2016). Methods for building course recommendation systems. In Proceedings of the 2016 International Conference on Knowledge and Systems Engineering (KSE 2016, ISBN 978-1-4673-8929-7, **IEEE Xplore. Scopus indexed.**

[CT2] Huynh-Ly Thanh-Nhan, Le Huy-Thap, and Nguyen Thai-Nghe. (2017). Toward integrating social networks into Intelligent Tutoring Systems. In Proceedings of the 2017 International Conference on Knowledge and Systems Engineering, (KSE 2017), ISBN 978-1-5386-3576-6, **IEEE Xplore. Scopus indexed.**

[CT3] Huynh-Ly Thanh-Nhan, Le Huy-Thap and Nguyen Thai-Nghe. (2020). Integrating courses' relationship into predicting student performance, International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE), pp. 6375-6383, Vol. 9, No 4, 2020. ISSN: 2278–3091. **Scopus indexed 2020 .**

[CT4] Huynh-Ly TN., Le HT., Thai-Nghe N. (2021). Integrating Deep Learning Architecture into Matrix Factorization for Student Performance Prediction. In: Dang T.K., Küng J., Chung T.M., Takizawa M. (eds) Future Data and Security Engineering. FDSE 2021. Lecture Notes in Computer Science, vol 13076. **Springer, Cham. Scopus Q2**

[CT5] T.-N. Huynh-Ly, H.-T. Le, and N. Thai-Nghe, (2023), Deep Biased Matrix Factorization for Student Performance Prediction, EAI Endorsed Trans Context Aware Syst App, vol. 9, no. 1, Apr. 2023.