Analyzing Learning Behavior of Student Persona toward Non-Negative Matrix Factorization

Dinh Thi Dong Phuong and Hiromitsu Shimakawa

Abstract—**Estimation of motivation and learning strategy of students is crucial for a teacher to engage them in programming. Let us consider a persona, which is a virtual student representing a student group similar in motivation and learning strategy to learn programming. Personas enable the teacher to predict student behavior during the programming education course. The paper proposes a method to figure out the weight each student belongs to a specific persona. It uses non negative matrix factorization (NMF) to decompose a matrix of portfolio, which is extracted from their real learning behavior, into the product of 2 matrices. A matrix represents the weight of each student belonging to certain personas. The other represents persona features. For the NMF, determining persona feature matrix is essential to achieve the good factorization. From the learning behavior of 66 students, we found that the trends of motivation features along the course, such as learning time, test score, submissions before deadline is good indicator for the feature matrix.**

Index Terms—**Programming course, motivation, learning strategy, portfolio, persona, non-negative matrix approximation.**

I. INTRODUCTION

In programming education, the teacher is required for supervision appropriate for each student. As the current situation, the teacher cannot give such supervision because he has little information on the motivation and the learning strategy of the student. To make the matter worse, unsuitable supervision may demotivate the student a lot. For example, suppose a student with an exploring mind to acquire new knowledge and skill. Though he asks a question to improve the quality of a code he has made to a teaching assistant (TA), the TA gives her the answer along with its demonstration, instead of some advice to improve the code. In this case, the supervision of the TA is unsuitable for her, because it results in degrading her intrinsic motivation. Even though she wants to explore the skill by herself, which is a good learning strategy to acquire programming skills, the TA deprives the chance to practice of her. It eventually prevents her from continuing the good learning strategy she owns. If the TA knows her motivation and learning strategy at the time she asks the question, he would never give inappropriate supervisions to her.

The paper proposes a method to know the motivation and the learning strategy of a student at any time during a programming course. It identifies personas with the contextual inquiry [1], which reveals learning experiences of

past students under specific contexts through interviews. The examination of interview results brings out groups of students similar in motivation and learning strategies. A persona [2] is assigned to each of the student group, to represent their typical learning behavior. Personas provide the supervisors with a clear image of their motivation and learning strategy. On the other hand, portfolios of students record their progressive learning behavior such as the learning time and scores. Obviously, the learning behavior depends on their motivation and learning strategies. The learning behavior of a student would be able to be presented with personas. NMF [3] would be the good way to factorize the portfolios into personas features and weights. The NMF requires the good initial personas features to achieve the correct factorization result.

II. STUDENT PROGRAMMING BEHAVIORS

A. Factors Determining Behaviors

R. E. Slavin [4] says that "motivation is internal process that activates guides and maintains behavior over time". A strong motivation to study programming would be a determining factor leading to the success of learning. Together with motivation, effective learning strategies are as important as the motivation to reach to achievements. MSLQ [5] is a good guideline to make questionnaires to understand components of motivation and learning strategies of a student in programming study. MSLQ not only provides descriptions of each component, but also enumerates sample questions.

For motivation, MSLQ lists up intrinsic, extrinsic, task value, expectancy, and affective components. Learners who have intrinsic components participate in target tasks because their goals are achievement of the tasks. They work because of reasons such as their challenge, curiosity, and mastery. For intrinsic components, MSLQ lists example questions like "In a class like this, I prefer course material that really challenges me so I can learn new things". Learners depending on extrinsic components participate in tasks because a learning task is the means to their goals. Components such as rewards, scores, and competition make them engage in the tasks. Task value components are determined by learner perceptions of the tasks in terms of interest, importance, and utility. Expectancy components refer to learner beliefs that their efforts will results in good outcomes. Affective components are related to anxiety for tests, grading for credits, and so on. For example, since they worry about tests, they make efforts.

A learning strategy is a personal approach to understand information and solve problems. Learning strategies consists of resource management strategies and self-regulation strategies. The resource management relates to the learner ability to understand the usefulness and effectiveness of

given resources for the learning. As for resource management, MSLQ enumerates study environment strategies regarding to setting of study places, peer learning strategies corresponding to collaboration with peers, and help seeking strategies to get supports from others including peers and instructors. Self-regulation strategies work in the process whereby learners systematically direct their thoughts, feelings, and actions toward the attainment of their goals [6]. Yukselturk [7] states self-regulation learning strategies are strong predictor for success in on-line courses. Self-regulation strategies in MSLQ involve cognitive strategies, meta-cognitive strategies, time strategies, and effort regulation strategies. Cognitive strategies refers to methods to learn, such as how to remember new concepts, to organize them, and to understand them, while meta-cognitive strategies correspond to methods to tune and adjust cognitive strategies. Time strategies plan their study time. Effort regulation strategies control their efforts in the face of distractions and uninteresting tasks.

B. Context

Some students are eager for programming, while others have no interest in it. Every learner has a certain state in a specific time. In a learning process, students receive stimuli in her education environment. Suppose a student who stays in a certain state. When she receives a stimulus, she takes an action against the stimulus, depending on the current state in her environment. Her state changes into a new one after the action.
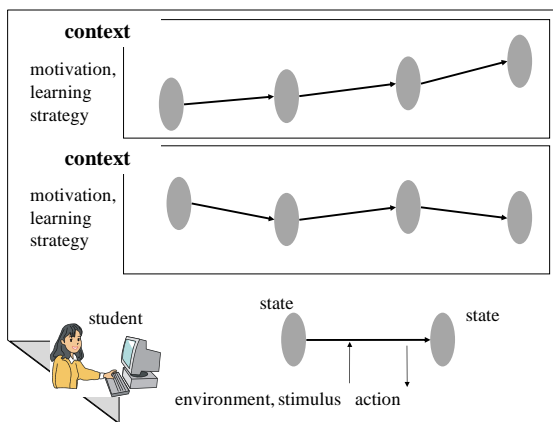

Fig. 1. Motivation and learning strategy shapes the context.

A context is a chain of states which change according to stimuli. Fig. 1 visualizes the concept of a context. Since the chain of states in a context results from certain motivation and learning strategy, the analysis of a context reveals them.

We use the contextual inquiry to get contexts in the learning of past students through interviews. The contexts are stated in scenarios which are outcomes of the interviews. Each scenario expresses many transitions of states of a student. There are many contexts described in a scenario. Classifying similar scenarios into several groups, we obtain motivation and learning strategies of students who belong to each of the groups. A persona is assigned to each group. A persona is a virtual student representing motivation and learning strategies of students in one group. MSLQ works as a good guideline to classify students from the view point of motivation and learning strategies stated in the scenarios.

C. Student Portfolio

Motivation and learning strategy are expressed to the outside through behaviors. Good facilitation tools for programming learning can record the behaviors of the students. The records indicate, for example, when a student studies programming, what assignments she solves, where she solves it, and the articles she refers to during the learning process.

Table I shows an example of a portfolio of 5 students from *a* to *e*. It represents 7 features of each student. The learning time is the time they make access to the programming practice support web site which gives assignments, sample codes, guidance, and so on. The submissions are the total number of source codes submitted to the servers. The compilation frequency is the average compilation times of a source code over a specific time period. The compilations are the average compilation times until they submit a source code. The viewing check-item indicates the times the student refers the check items, which show explanation and guidance for students to achieve a good source code.

Obviously, if a student works hard to achieve good source code, her learning time would be long. If a student has a willing mind of challenge and exploration, she would solve many assignments. If a student spends little time to make source codes for difficult assignments, the codes would not be the good one, which gets low score. Responses to stimuli in specific contexts depend on the motivation and learning strategies of the students. Progressive learning behavior represented in the portfolio does represent motivation and learning strategy of each student.

TABLE I: PORTFOLIO OF STUDENTS

X	Learning time	Submissions	Compilation frequency	Compilations	Viewing check-item	Referring samples	Score
a	71.97	32.17	1.25	16.8	4.89	5.0	5.7
b	110.01	38.96	1.15	26.4	4.47	7.0	8.2
c	113.78	41.78	1.3	27.2	5.06	5.06	8.9
d	92.6	30.27	0.78	22.4	3.02	4.02	5.5
e	49.01	36.66	1.95	10.4	7.67	.67	4.2

D. Non-Negative Matrix Factorization (NMF)

Let a non-negative matrix X, where each row contains the values of *n* features of one student, represent a portfolio of *m* students. The non-negative matrix factorization (NMF) [8] seeks to decompose X into the product of 2 matrices, W and P: X, W * P,

where every element in an $m * r$ matrix, W, and an $r * n$ matrix, P, is non-negative. We refer to W as a weight matrix, while P as a feature matrix. We can achieve the exact decomposition of X into W and P in specific conditions [3]. We can also get the good approximation of W and P, using the hill climbing method.

Since all elements in X, W, and P are non-negative, we can regard the matrices as follows. As stated above, the portfolio of a student represents her motivation and learning strategy. Each of *r* rows of the feature matrices, P, presents the values of *n* features of a specific group, where *r* is the number of groups. Each of *m* rows in the weight matrix, W, represents

the weight for a specific student to belong to each of n groups, where m is the number of students appearing in the portfolio.

III. PERSONA ESTIMATION OF A STUDENT

Fig. 2 presents the method to find out the weight for a student to belong to groups in the current programming course. In the current course, we obtain a portfolio of every student from the beginning of the course to the current time. We make matrix X from the portfolios of m students. Decomposing X, we figure out the weight matrix, W, after we achieve the feature matrix, P.

To achieve matrix P, interviews for the contextual inquiry are conducted for the students of the last programming course. From the interview result of each student, motivation and learning strategies of the student are described in a scenario. Referring to MSLQ, scenarios similar in motivation and learning strategy are put into the same group. The procedure results in a small number of scenario groups. We assign a persona to each of the group. A persona represents a virtual student to represent the group. Each persona has a virtual scenario, which integrates the motivation and learning strategies of a group. Each row of P represents characteristics of motivation and learning strategies of the persona. P represents the features of personas of the last course.

A. Scenario

In the proposed method, the classification of student groups is essential to achieve a good decomposition of a portfolio matrix into the product of a weight matrix and a feature matrix. Especially, features of motivation and learning strategies to distinguish each group play vital roles. They not only work as criterion for the classification, but also represent the degree of each student in portfolio belonging to them.

With questionnaire methods, it is hard to get real and practical hints for the classification. Since the students who give answers to the questionnaire may not have real experiences for conditions assumed in questions, questionnaire methods prevent collection of truthful information from learners. On the other hand, the contextual inquiry is a good way because the students join the interviews freely talk about their real experiences, from which, real learning contexts are expressed. The contexts reveal hints for features.

The contextual inquiry method insists to delve into learning processes of the students with interviews among themselves. Every detail of behaviors of students in specific conditions is clarified as an interview goes on. Suppose the student mentions help from her friend to solve the assignment, when an interviewer who experienced the same programming course, asks how she solves an assignment. The interviewer asks why the help occurs, how the help takes place, and how the interviewee feels after the help. The further the interviewer delves into the context, the clearer image of the learning process the interviewer obtains. A scenario summarizes what the interviewee has experienced. The scenario expresses contexts, which consist of sequences of actions and feelings toward maters occurring in the learning process in specific conditions of the student.

B. Persona

As Cooper states, scenarios of learners similar in their characteristics can be compiled into a scenario for a persona. A persona is a virtual learner with a scenario stating its behaviors in specific contexts. After obtaining scenarios of contexts, we know motivation and learning strategy of every student, referring to MSLQ. Scenarios similar in motivation and learning strategy are put into a group. A persona represents motivation and learning strategy of a group. A scenario of a persona presents typical learning contexts of students corresponding to the persona.

Personas and their scenarios make the designer of programming course grasp a single image of members of a student group in advance. In the course time, if the supervisor knows what persona the student belongs to, or in other words, her motivation and learning strategy in a specific context, the supervisor would give more appropriate supervisions to the student.

Moreover, the number of students join the course are often very big, because programming courses are usually compulsory ones in computer science education programs. The students are divided into separate classes where a teacher and some teaching assistants take in charge of supervisions. It causes the variance of teaching among the classes. The variance should be avoided to bring a fair education to all of the students taking the course. In each class of about 40 students, the diversity of the students is also the mater preventing the persons in charge from giving a fair education. Persona contributes minimize the variance among the classes as well as covering the diversity of students in each class.

C. Meanings of Values in Matrices

Supposed there are two personas, P1 and P2, in total. Features characterize the personas are shown in matrix P. Let us take sample data as described in Fig. 3. The values of matrix P indicate the features of the two personas.
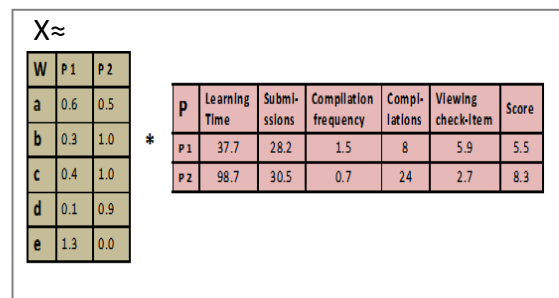

X≈									

W	P1	P2
a	0.6	0.5
b	0.3	1.0
c	0.4	1.0
d	0.1	0.9
e	1.3	0.0

*

P	Learning Time	Submissions	Compilation frequency	Compilations	Viewing check-item	Score
P1	37.7	28.2	1.5	8	5.9	5.5
P2	98.7	30.5	0.7	24	2.7	8.3

Fig. 2. Weight matrix and feature matrix.

P1 would represent the persona who is easy-going to the learning than P2 because P1 does not spend much time to solve assignments as P2 does. Moreover, the compilations are few, but the compilation frequency is high. It means P1 does not think carefully in the learning process. P1 also often views check-items to follow the guidance to finish the source codes quickly without careful considerations to achieve good source codes. Even though P1 has many submissions as P2 has, P1 gets low score for the source codes. On the other hand, P2 spends much more time to solve assignment. P2 has high compilation times. P2 also has many submissions. P2 would be the persona who likes to study programming, and wants to explore knowledge and get skills. Moreover, even P2 has

long learning time and many compilation times, P2 has small compilation frequency and small viewing check-item times. P2 would be a carefully thinking persona for achieving good source codes.

The values in matrix W represent the weight of a student to a certain persona. With the given sample data, student ***b, c, d*** have strong tendency to belong persona P2, because their values are 1.0, 1.0, and 0.9, respectively in the column of P2. Student **e** would not belong to P2 because his value corresponding to P2 is 0.0. Student **e** would belong to P1. Student **a** may belong to both P1 and P2 because the weights corresponding to P1 and P2 are 0.6 and 0.5 respectively. A student may belong to only one persona, while another may belong to many personas.

IV. PERSONA FEATURE MATRIX

A. Obtaining Persona

The contextual inquiry method is used to get scenarios expressing learning contexts of the students. We let students having finished the last programming course conduct interviews with each other to describe their contexts. The good time for the interview is at the beginning of right next semester after the semester of the programming course. The students have enough break after the programming course. If we ask them to do the interview right after the programming course, many complaints would fill up the interviews because the students have to make a lot of efforts in the programming course. If the interviews are conducted after a too long break, they would lose truthful contexts.

The inquiry topic is programming motivation. Following is one snippet of one interview of student R and student E.

R: What makes you incline to programming?
E: I regard programming as a challenge. When I have solved a tough assignment, I get a strong sense of achievement.
R: When you face a tough one, you might sometimes find no way to solve it. Have you ever run into such a situation?
E: Yes, many times. In such a case, I will repeat to check sample codes in the textbook. Sometime, I try easier assignments in the same section.
R: What will you do, if you cannot get anything from them?
E: I will search Web pages explaining similar matters using the Internet. I do not prefer to be supervised by TAs, because I feel lost in the challenge.
R: But, they give you clues, even an encouragement, sometimes.
E: I am not pleased, even if they encourage me. But, when they give high grades to my codes, I get satisfactions.

We create scenarios from the interviews of 73 students. The scenarios are classified using MSLQ. The result reveals 3 groups of students similar in motivation and learning strategies. One group consists of 27 students willing to learn. They understand the more they engage in the study, the more they can get knowledge and skills. Some of them want their high ability to be praised by supervisors or friends. This group, represented by persona challenging persona, is specified to have strong intrinsic, expectancy, and task value components, along with weak extrinsic ones. Moreover, challenging persona knows how they should study to improve their abilities. They know to utilize various

resources such as explanations by TAs and sample codes in the Internet. It reveals that challenging persona has learning strategies to achieve a good result.

The second group consists of 19 students who understand the compulsory programming subject brings many benefits to them. Their learning purpose is to get a credit. Their attitudes to the learning are not stable. As far as they can go on with assignments easily, they are eager to study. But, for assignments which seem hard to solve, they tend to put them aside, because they do not believe their ability. They often stick to specific resources such as TA helps. They do not know utilizing many kinds of resources. We refer to this group as persona easygoing persona. We regard easygoing persona has weak intrinsic, and strong extrinsic, weak task value, and strong affective components. Easygoing persona has no learning strategy.
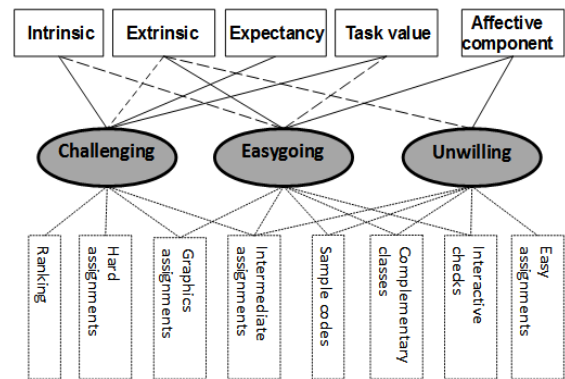

Fig. 3. Persona and hints from contextual inquiry.

The third group, represented with persona unwilling persona, consists of 27 students who think that they are obliged to learn programming. They do not like to learn but have to, because of the compulsory subject. They learn only at school with help of supervisors. Different from challenging persona and easygoing persona, unwilling persona does not show any will to learn. Unwilling persona has no intrinsic component. Since unwilling persona minds obligations so much, unwilling persona has strong affective component and weak extrinsic one. Unwilling persona has no learning strategy.

Fig. 3 shows associations of the three personas with motivation components. The solid lines show strong associations, while dashed lines correspond to weak ones. The strength of the association is manually determined when course designers determine personas from scenarios of many learners, referring the combination of sample questions and their correlation with the final grade in MSLQ.

Fig. 3 also indicates service references of the personas in their summary learning behaviors with dotted lines. Assignments are requirements for students to make source codes. Interactive check is interactive supervision of a TA and the student, where both of the TA and the student examine the source code of the student to confirm what the student understood. Complementary classes are prepared for teachers and TAs to supervise students at a loss to solve assignments. Sample codes are model codes with detail explanations. Graphics assignments enable students to view execution results visually. Ranking shows scores of top ranked students.

B. Persona Feature Matrix

The contextual inquiry of students of the previous course brings personas in the students. Our purpose is to understand the personas of a student in the current course. With MNF factorization, it is essential to find initial good values of features of personas. To find the initial persona feature matrix, we use the result of the questionnaire which is conducted at the end of a course of 68 students. There are 8 items with yes/no options in the questionnaire relating to motivation and learning strategies as following:

1. I want to challenge difficult programming exercises.
2. Most important thing in the programming exercise would be to take a good score.
3. I've studied programming because it is necessary for my future works.
4. If I study properly, I am sure to obtain programming abilities.
5. I have a sense that I am improving programming skills in the course.
6. Every week, in the class, when the test time finishes, I always suspect whether others get better scores than mine.
7. I always think carefully what I can achieve in each exercise.
8. I try to accomplish even assignments I am not interested in.

Item 1 strongly represents the challenging persona. Item 2 strongly represents the unwilling persona. We divide the questionnaire result as the decision tree in Fig. 4. As a result, there are 42, 20 and 6 students belong to challenging persona, easygoing persona, and unwilling persona, respectively. The interviews using contextual inquiry method of the students brings hints for us to analyze their behavior data taken during the course. Students belonging to challenging personas mention of long time learning, solving all assignments, seeing ranking page. The students of easygoing persona states that they solve only easy assignments, want to know score right after the test, are anxious of hard deadline. The ones of unwilling persona state that they study only within the compulsory class time, see only easy assignments.

Corresponding to the hints, we extract the learning time, the total source code submissions, number of total clicks, number of clicks on ranking button, the submission for easy assignments, number of clicks on score page, score of small test, submissions in the day just before the deadline from the behavior data of the students.
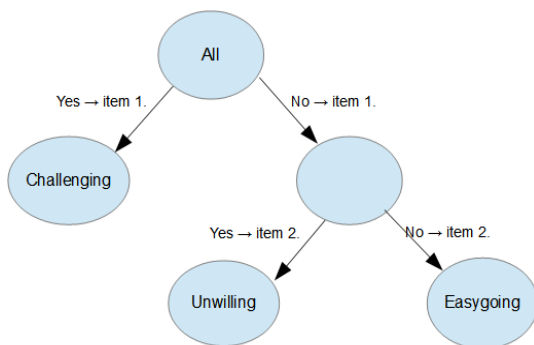

Fig. 4. Personas from questionnaire.

Among the above features, learning time, total clicks, small test score, and submissions before the deadline are noticeable. Fig. 5, Fig 6, Fig 7 and Fig. 8 show the result.

Until week 9th, learning time, total clicks and small test score of challenging persona is almost higher than those of easygoing persona. Those of easygoing persona are also higher than those of unwilling persona. Contrary, last day submissions of easygoing persona and unwilling persona is higher that of challenging persona.

More noticeably, it is the trend of the bars of the three personas. In learning time, the bars of challenging persona are very high at the week 2th, 3th, and 4th. The bars get lower in the week 5th and 6th and get high at week 7th. After week 7th, the bars keep horizontal. Whereas the bars of easygoing persona and those of unwilling persona do not get high at the week 7th.
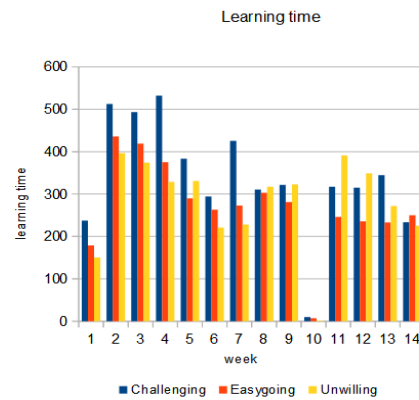

Fig. 5. Learning time of personas.

Before the and after the mid-term test, learning time of unwilling persona get highest suddenly. After the mid-term test, last day submissions are higher than ever before.

To small test score, students of challenging persona tend to get full scores along the course. Students of unwilling persona do not get full scores most of cases. Ones of easygoing persona get both full and not full scores.
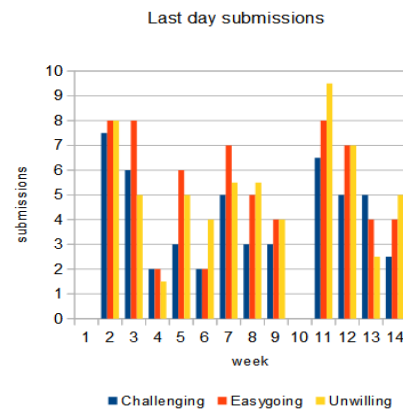

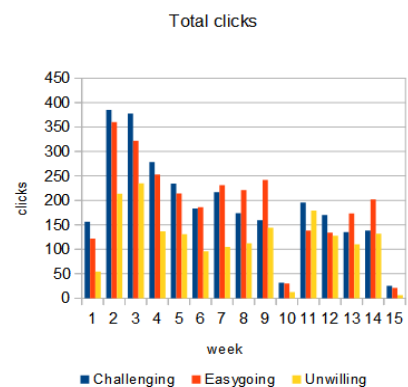Fig. 6. Last day submissions of personas.

Fig. 7. Total clicks of personas.

The analysis shows that, to understand persona of students at a certain week, we can use behavior data collected until the week. Suppose we are now in week 10th. We can use behavior data from week 2 to week 10.As the graphs show, the challenging persona decrease its learning time from week 5th, (except week 7th). Maybe, it comes from their industrious learning in the early weeks. As the course schedule described in Table II, at week 7th, the learning time of challenging persona gets high because the content nested *loop and debug* seems to be hard. Since challenging persona work so hard in the early weeks, they have already acquired basic programming skills. That's why they do not have to work hard in the later weeks.

On the contrary, the unwilling persona works more before the mid-term test, because it minds a poor score in the test. In addition to that, it works hard just after the mid-term test, because its poor score in the test has surprised it. The unwilling persona starts to study hard, being anxious about a failure in getting the credit. Since the easygoing persona does not care the mid-term test, its learning time and clicking behavior do not change around the mid-term test.
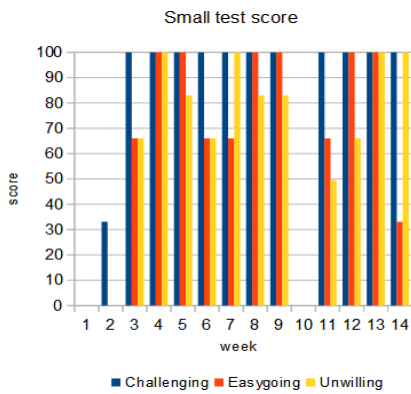
Fig. 8. Small test score of personas.

The consideration in the above implies, when we want to identify a persona of specific student in a later week, we should consider the trend of his behavior data from the beginning the the semester to the week, instead of his behavior data in the weeks near the current one.

TABLE II: COURSE SCHEDULE

Week	Content	Week	Content
1	Linux and Emacs	9	Function with array
2	Input and output	10	Mid-term test
3	Variable & expression	11	String
4	Conditional statement	12	String with pointer
5	Loop statement	13	Structure
6	Nested loop, debug	14	Recursive call
7	Function	15	End-term test
8	Array		

V. CONCLUSION

The paper aims at estimating personas of a student in the current week in a programming course. Contextual inquiry is used to get scenarios expressing motivation and learning strategy of students for personas. Learning support Web site if to take real behaviors along with the current course. MNF factorizes the portfolio which is extracted from the behaviors, into the weight and the persona features. Analyzing the portfolio of 66 students, we found that the trends of features are good indicators to determine initial persona feature matrix.

REFERENCES

[1] H. Beyer and K. Holtzblatt, *Contextual Design: Defining Customer-Centered Systems*, Morgan Kaufmann, 1998.
[2] A. Cooper, "The Inmates are running the asylum: Why high tech products drive us crazy and how to restore the sanity," *Pearson Education*, 2004.
[3] D. Donoho and V. Stodden, "When does non-negative matrix factorization give a correct decomposition into parts?" in *Proc. Conference of Advances in Neural Information Processing Systems*, vol. 16, 2003.
[4] R. E. Slavin, *Educational Psychology: Theory and Practice*, 10th Edition, Pearson, p. 317, 2006.
[5] P. R. Pintrich, D. A. F. Smith, T. Garcia, and W. J. McKeachie, "A manual for the use of the Motivated Strategies for Learning Questionnaire (MSLQ)," National Center for Research to Improve Post Secondary Teaching and Learning, 1991.
[6] D. H. Schunk and B. J. Zimmerman, *Self-Regulated Learning: From Teaching to Self-Reflective Practice*, Guilford Press, 1998.
[7] E. Yukselturk and S. Bulut, "Predictors for student success in an online course," *Educational Technology & Society*, vol. 10, issue 2, pp. 71-83, 2007.
[8] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, Oct. 21, 1999.

Dinh Thi Dong Phuong was born in Quang Nam, Vietnam in 1979. She received her bachelor of mathematics and information science in 2001 in Danang Education University and received the master of information engineering in Ritsumeikan Universtiy in 2008 in Japan. From 2009, she has been a PhD candidate at Ritsumekan.

Ms. Dinh Thi Dong Phuong is currently interested in education engineering.

Hiromitsu Shimakawa was born in 1961 in Japan. He received Ph.D degree from Kyoto Univ. in 1999 and joined Ritsumeikan Univ. in 2002.

Currently, he serves as a professor in Ritsumeikan University. His research interests include data engineering, usability, and integration of psychology with IT.

Professor Hiromitsu Shimakawa is a member of IEEE and ACM.