Applying Inverted Classroom to Software Engineering Education

Eun Man Choi, *Member, IACSIT*

***Abstract*—A successful engineering education must possess a wide engineering disciplines and real-world skills. Software engineers have ability to apply several methods in practice. Engineering educators face a complementary and more challenging problem, how to effectively teach students about the whole software engineering process. An inverted classroom can be a solution to give the educational environment that mixes the use of technology with hands-on educational activities. Traditional instructor-led lessons are replaced by VOD using the time outside of the classroom. Lecture time was devoted to active learning activities that can be done such as design studio and game storming. For case study inverted classroom concept was applied to introductory software engineering course, and measured by controlled experiments. The experimental group was found to be effective in two aspects. Fast feedback through well made design heuristics of learning teamwork and good influence on the overall schedule for the project was.**

***Index Terms*—Case study, inverted classroom, measurement of training effect, software engineering education.**

I. Introduction

If software engineering education becomes a live training, learned principles should be able to be applied to work together as developers, and the relevant parties to create software. Education for software engineers in the future needs to be mobilized by the various efforts in order to make active learners in working environments as well as classes. Traditional isolated software engineering education is not effective for training creative software engineers, because students learn passively and have no chance to learn by doing. Co-operation with other learners and mentoring students work should be able to get.

Inverted classroom is a training environment to use both simple educational activities and lectures [1]. Inverted classroom lessons from traditional instructor-led training in engineering education for small activities are replaced by VOD using the time outside of the classroom, instructor-led training was done. 3 hours of lecture time was devoted to active learning activities such as a design studio, gamestorming [2]. Learning outside of the classroom was done for at least three hours to watch videos.

Lecture in inverted classroom concept can be implemented by VOD, download video iPod broadcasting, and podcasting. Several cases use the iPod on education by inverted classroom. However, the most effective multimedia for education with inverted classroom [2], [3] has not been discussed and measured.

Learning activity in inverted classroom takes place both inside and outside the classroom. Passive learning takes place mostly outside the classroom by listening and watching VOD. Lectures in classrooms are filled with activities for software analysis, design, and operation of an in-depth conversation and mentoring.

Most of the programming exercise in computer science courses requires the coding and thinking by oneself for the small problems. The other hand, requirement analysis, design, testing principles for software engineering education need to be practiced by simple activities beyond the level of understanding principle and method. For example, teamwork-building, software features, use cases in a class project should get fast feedback to find it right and learn effectively.

This research applies inverted classroom concept to software engineering education in order to measure the effectiveness of these motives. For the controlled experiment, one of the two courses was conducted in the traditional way with software development projects. The other course conducted both watching pre-VOD before classes and in-class activities as the active learning session by accepting the concept of gamestorming, a structured activity that provides scope for thinking freely, even playfully. The learning achievement of students who have taken and the results of the project were compared to evaluate and analyze the efficiency of inverted classroom in software engineering education.

Chapter II describes general educational model and related research. Chapter III presents the inverted classroom model in software engineering lecture and learning activities. In Chapter IV, the proposed method and results from controlled experiments are covered. Chapter V includes conclusions and future research challenges.

II. Research Background

Training for software engineers can vary greatly depending on where the emphasis is. Theoretical methods for analysis, design, coding, testing, maintenance, planning, management, can be covered only if using traditional teaching methods like lecturing. But it has shortage to cultivate the student’s ability to apply theory to real-world problems as engineers, and solve problems through collaboration to develop real-world software. This chapter introduces the educational model to use for software engineering education and related research.

A. Collaboration Models

Personnel working in the field of IT are responsible for large, complex software system requirements, design, implementation, testing, installation, and operation. These tasks cannot be done by working alone, but done by

Manuscript received February 27, 2013; revised April 30, 2013.

Eun Man Choi is with Department of Computer Engineering, Dongguk University, Seoul, 100-715, South Korea (e-mail: emchoi@dongguk.edu).

DOI: 10.7763/IJEEEE.2013.V3.205

collaborating with stakeholders and colleague. In addition, methods and theories for each of these tasks are important to work in a software project, but the other soft skills such as group work communication, collaboration, organization, decision-making, conflict resolution, leadership, social, concentration, etc. are necessary too. Traditional lectures and hands-on labs have limitation to learn these skills for collaboration.

Cooperative education model [4] provides project team and the co-op to simulate future business cooperation by providing several educational activities, such as design studios. Educational models for collaboration [5] needs the ceaseless communication between instructors and learners. The communication should take place on a specific topic, and suggests that to happen evenly and diverse communication between group members.

Inverted class was designed to follow the characteristics of the cooperative education model in this study. This paper introduces the technology and summarizes the results of educational practices.

B. Related Research

There have been closely related to this research study such as Day and Foley [6], Kaner and Fiedler [7], Gannod and Burge, Halmick [8]. Inverted classroom research in the field of HCI (human computer interaction) of the Day and Foley was introduced to control the difference between the level of learning courses and experimental courses. Kaner and Fiedler's research in software testing lecture to introduce the experience of applying the inverted classroom approach. Gannod and Burge’s research and Halmick’s research on various subjects of computer science measured effectiveness of inverted class through surveys for operating after effects are.

If you look at the details of related research it has small clue and more difficult to find out software engineering practices about the design, content and teaching and learning model in the group learning activities. Inverted classroom operating results and effect analysis have not been able to find out.

III. Design of Inverted Classroom

Generation Y, called 'millenial', has a lot of impact of the Internet and SNS when they were growing up and the very different tendencies from Generation X, called baby bummer. According to Frand [9] the Y generation is very natural use of the Internet which is used more than TV, and prefer that the image or video. They also think the more important skill would not be simply build the knowledge but dealing with complex and ambiguous information. Generation who grew up with the Nintendo Game accustomed to learn by trial and error rather than the logic. They also used to multi-tasking. Generation Y students cannot tolerate the slow response and always wants to be connected by the media such as SNS.

According to Foreman’s research [10] an ideal educational environment must provide customization and immediate feedback by students and built out conceptual structure which should be able to be maintained. Students to want to actively explore and established knowledge can remember in the long run it through various activities and are able to apply them in the production.

This paper provides an ideal educational environment for software engineering education, reported and analyzed its effects.

A. Educational Model

Traditional education model consists of students meeting regularly in the classroom and teacher centered lecturing to pass one-sided knowledge. Regular off-line classroom lectures has limit on the time and the number of passing knowledge. The biggest drawback of traditional education is that active learning does not occur. Problem solving activities should be added to traditional education model.

Active learning-centered education model can be achieved in the process of cooperation to focus as a group on a particular topic, and given challenges to participate. However, collaboration-centric tasks that rely solely on education may not be able to achieve the intended learning objectives. Therefore, to some extent, instructor-led training needs to be mixed.

Inverted classroom is a flip-over that traditional instructor-led training is moved to outside of the classroom and active learning for cooperative task-oriented activities is done within the classroom. For example, concepts and methods of software engineering process, planning, requirements analysis, design, testing are passed through the asynchronous medium. Faster feedback to requirements analysis and design software from the actual work activities by students will be provided by instructor when they meet in the classroom. In that moment, active learning should be done.

The traditional education model and inverted classroom are compared and summarized in Table I.

TABLE I: The Arrangement of Channels

	INSTRUCTOR		STUDENT	
	Traditional	Inverted class	Traditional	Inverted class
Preparation of lecture	Standard preparation time	Standard prep plus prep for additional material, and produce recorded lecture	None	None
Preparation for class	Same as lecture preparation	Develop learning activities	Readings	Readings, view DVD
Attendance	None	None	Only if required	Required
Learning activities	Instructor feedback delayed, contact and guidance limited office hours	Instructor feedback in process, contact during entire contact hour.	Outside class	In class/ outside class

Inverted classroom is different from the traditional educational model based on the perspective of instructors. Inverted classroom has advantage of more opportunity for students to interact with. Even the most interactive lectures are likely to actively involve only a subset of the students. In the inverted classroom, the instructor works directly with individual students during contact hours. Most of the class time in inverted classroom model can be spent with these students that are struggling, as opposed to the traditional lecture where most of the questions posed during discussion come from the stronger students. Instructor and student interact together to solve problems and participate in the inverted classroom mentoring so that all students can interact evenly.

In-class activities make engagement with not only the learner but also the instructor. The instructors often lose their interest while the instructors repeat many times in the traditional lecture. One sided conversation for the same content might be vanish and less interest to the instructor. Inverted classroom activities have open problem so that there is no fixed solution. In each semester, instructor encounters other students who confront other problems and more interest to teach.

The inverted classroom provides an easy way to involve guest speakers in classroom instruction. In a class that covers a broad subject area, such as software engineering, not all instructors will be equally adept at all topics. In addition, it may be beneficial to bring in some “outside voices” such as bringing in experienced industry professionals. Trying to schedule guest speakers for a traditional lecture can be difficult. With the inverted classroom, speakers can deliver their portion of the lecture at their convenience.

Students can learn in more active way to participate learning activities beyond the passive learning from lectures. Students, when engaged to the activity, can gain more chance to get knowledge and experience because this method is appropriate to new generation of students. New generation students can involve in several tasks to multi-tasking at the same time so that they accustomed to video at any time, then stop studying can be used for other tasks.

The best thing of the inverted classroom is quick feedback. Class activities instead of one-sided lectures make instructor face-to-face mentoring results of analysis, design and giving a fast feedback about other activities. Students can understand the principle of work and know any fault or lack of work right away. A wide variety of video can make customized training possible depending on the learner's skills or interests. Video with a mixture of the blog or SNS is exposed to a variety of learning environments than students in the classroom listening to the one-sided.

Iterative learning by watching lecture for inverted classroom videos also forms the students' long-term memories in their brain. That provides greater learning effect and leads to adaptability for practical skills.

B. In-Class Activities

There is a big difference in the scope and depth of the learning activities between traditional lecture and the inverted classroom lectures as shown in Table II. The number of assignments, when using the inverted classroom model, can be much higher than in the traditional classroom model. Traditional lecture has complex challenges so that needs the results of several learning objectives. While the inverted classroom impose challenges to setting goals and specific classes of simple activities that are available here to get immediate feedback and simplifies the measurement of training effects. With respect to assignment depth, learning activities in the inverted classroom model, due to time constraints contain less depth than in the traditional model.

The inverted classroom activities for software engineering courses were designed and run as shown in Table III. Learning outside the classroom is provided by VOD described by theoretical methods for analysis, design, coding, testing, etc. After watching VOD students will take part in actual software development activities in class such as planning, analyzing, designing, coding, and testing.

TABLE II: COMPARISON OF LEARNING ACTIVITIES

	Traditional	Inverted
Number of assignments	low	high
Outcome coverage	low	high
Feedback	delayed	Immediate
Depth per assignment	high	low

TABLE III: LEARNING ACTIVITIES BY INVERTED CLASSROOM FOR SOFTWARE ENGINEERING

Week	Out of classroom	In classroom
1		Lec 01: Introduction
2	DVD01: Process	Act01: Team building - MBTI - Case study for process
3	DVD02: Requirement analysis	Act02: Conceptualization - Project concept - Mission statements
4	DVD03: Requirement spec	Act03: Post-it gamestorming - Brainstorming for feature list
5	DVD04: Functional modeling	Act04: Use-case analysis - Write use cases
6	DVD05: Static modeling - Find classes - Class diagram	Act05: Class diagram design studio - Pick up class candidates - Identify relationships - Draw class diagram
7	DVD06: Dynamic modeling - Sequence diagram - Communication diagram	Act06: Sequence diagram design studio
8	DVD07: Design principles - Abstraction, modularization	Act07: Finding quality characters - Applying design principles
9	DVD08: UI design - UI design elements - UI design case study	Act08: UI/UX Design studio
10	DVD09: Design patterns	Act09: Applying design patterns
11	DVD10: Implementation - Coding style - Refactoring	Act10: Writing coding standards
12	DVD11: Testing(1) - Black-box testing	Act11: Writing test cases for equivalence partitioning
13	DVD12: Testing(2) - White-box testing - System testing	Act12: Analysis of execution paths and writing test cases

One good example of in-class activities in Table III is post-it game brainstorming to find feature list [11]. Post-it cards are given to all members of the team attended. They think about what features can provide. Then, each member of project team writes a feature to be included in new system onto post-it. After a certain amount of time to write system features on post-in are shared to colleagues by explaining their ideas. Good idea from another team during explanation can be added to their post-it description. Features after explaining the entire team gathered are pasted on board and organized by categories.

IV. EXPERIMENTS AND RESULTS

Controlled experiments are designed and performed to measure the effectiveness of inverted classroom in software engineering education. Two different types of software engineering in computer engineering major are opened in parallel. Inverted classroom lecture was opened in the control group and the experimental group was the traditional lecture method lecture. Population of the control group and the experimental group is shown in Table IV. Student population were random not to affect other elements.

TABLE IV: POPULATION OF TWO GROUPS

	Male	Female	Junior	Senior	Total
Inverted classroom	23 (61%)	15 (39%)	33 (87%)	5 (13%)	38
Traditional	24 (69%)	11 (31%)	29 (83%)	6 (17%)	35

Lectures, assigned project, and assessment for the control group and the experimental group are same. The independent variable for controlled experiment is watching video lectures and taking part in lessons to apply various principles and methods of cooperation centered activities.

The dependent variables are comprehension level of the course content, project progress, project feedback, satisfaction, teamwork, written test scores, project scores, and lectures satisfaction.

For analyzing the result of experiments, the following hypotheses are set whether to adopt and observe samples examined.

[Hypothesis 1] The experimental group understands the course content better than the control group.

[Hypothesis 2] The experimental group meet project schedule and progress more accurate than the compared control group.

[Hypothesis 3] The experimental group can be great feedback from professors and teaching assistant and satisfy better than the control group.

[Hypothesis 4] The experimental group has better project teamwork than the control group.

[Hypothesis 5] The experimental group has better written test scores than the control group.

[Hypothesis 6] The experimental group has better project score than the control group.

[Hypothesis 7] The experimental group has higher total lectures satisfaction than the control group.

The null hypothesis and the alternative hypothesis against the parameters of the population is set up the following form.

H0: ui = ut (Comprehension level of students in inverted classroom is same to traditional lecture courses student’s understanding)

H1: ui> ut (The student's understanding taking inverted classroom is higher than the traditional lecture course students’ understanding.)

The mean, variance, and the test statistic of the sample surveys shown in Table V required for the test.

If the significance level for T-test is set to 5%, P value of hypotheses 1-4 is less than .05 so that the null hypothesis is rejected. Students in inverted classroom group are a little more deep understanding about the course content. The deep understanding in inverted class comes from repeated watching lecture video and various activities of the class for applying best practices, because it is judged to be remembered in the long-term brain.

TABLE V: MEAN, VARIANCE, TEST STATISTICS

Classification		Mean	Variance	T-test	P
Hypothesis 1	Inverted	4.14	0.86	9.396	0.000
	Traditional	3.06	0.89		
Hypothesis 2	Inverted	4.24	0.83	5.831	0.000
	Traditional	3.72	1.06		
Hypothesis 3	Inverted	4.17	0.78	3.174	0.003
	Traditional	3.94	0.83		
Hypothesis 4	Inverted	4.32	0.64	5.459	0.000
	Traditional	3.67	1.09		
Hypothesis 5	Inverted	54.89	11.92	0.438	0.67
	Traditional	53.48	14.12		
Hypothesis 6	Inverted	182.25	13.77	0.469	0.32
	Traditional	184.23	19.96		
Hypothesis 7	Inverted	4.28	0.61	1.138	0.262
	Traditional	4.37	0.63		

The project schedule in inverted classroom group is well underway in the group activities compared to the traditional lecture group due to automatic regular progress check in the classroom. The other hand, the traditional lecture group has free group meetings or progress management cooperation in cyberspace guess that is not done well.

The student’s satisfaction on feedback in Inverted classroom group, of course, is higher than that in traditional lecture group. Because students in inverted classroom group take participation in the comments on the activities of the classroom professor has. Reason for high satisfaction in the project teamwork was a group of inverted classroom can say that offline meetings have been made in this class on a regular basis.

There was no significant difference in written exam grades, project evaluation score, and the lectures satisfaction between the control group and the experimental group. Because the contents of the written exam are focused on the software engineering theory, analysis, design, testing, and maintenance methods, rather than project practice. The score of project deliverables and lectures satisfaction did not differ significantly. That seems to come from the burden to prepare a variety of activities in the classroom and watching movies.

V. CONCLUSION

In this paper, applying the concept of inverted classroom to teach software engineering courses is compared to the traditional instructor-led training. Educational effects are surveyed by using T-testing. Inverted classroom replaced a passive learning to a variety of active group activities was done in time to meet professors and class mates.

The learning outcomes of the two groups were compared after a 16-week lecture. Inverted classroom student group better understands the content of the subjects from teamwork and project work so that could be tailored to a fixed schedule. Also fast feedback was conducive to learning. The other hand, handwriting test scores and teaching satisfaction could have a significant difference on effect. In conclusion, the concept of mentoring of inverted classroom effects educating several software engineering skills and the ability to project progress through face-to-faced education during activities.

In the future inverted classroom videos should be improved so that students can focus more in class activities and planning. A benefit is expected from a greater variety of learning and having fun by applying inverted classroom concept.

REFERENCES

[1] M. J. Lage, G. J. Platt, and M. Treglia, "Inverting the classroom: A gateway to creating an inclusive learning environment," *Journal of Economic Education*, vol. 31, no. 1, pp.30–43, winter 2000.

[2] G. Campbell, "There's something in the air: podcasting in education," *Educause Review*, vol. 40, no.6, pp. 32–47, November/December 2005.

[3] Apple. (October 2007). Apple in education: iTunesU. [Online]. Available: http://www.apple.com/education/itunes-u.

[4] C. Cheong, "From group-based learning to cooperative learning: A meta-cognitive approach to project-based group supervision," *Informing Science: the International Journal of an Emerging Transdiscipline*, vol. 13, 2010.

[5] S. W. Dietrich and S. D. Urban, "A cooperative learning approach to database group projects: integrating theory and practice," *IEEE Transactions on Education*, vol. 41, no. 14, 1998.

[6] J. A. Day and J. D. Foley, "Evaluating a web lecture intervention in a human-computer interaction course," *IEEE Transactions on Education*, vol. 49, no. 4, pp. 420–431, november 2006.

[7] C. Kaner and R. L. Fiedler, "Inside out: A computer science course gets a makeover," in *Proc. the Association for Educational Communication and Technology International Conference*, October 2005.

[8] G. C. Gannod, J. E. Burge, and M. T. Helmick, "Using the inverted classroom to teach software engineering", in *Proc. International Conference on Software Engineering 2008*, pp.777-786, 2008.

[9] J. L. Frand, "The information-age mindset: changes in students and implications for higher education," *EDUCAUSE Review*, vol. 35, no. 5, pp. 15–24, September–October 2000.

[10] J. Foreman, "Next-generation educational technology versus the lecture," *EDUCAUSE Review*, vol. 35, no. 5, pp. 12–22, September/October 2003.

[11] D. Gray, S. Brown, and J. Macanufo, *Gamestorming: A Playbook for Innovators*, Rulebreakers, and Changemakers, O'Reilly, 2010.

Eun Man Choi was born in Seoul, S. Korea in 1958. He received B. S degree in computer science from Dongguk University, Seoul in 1982 and M. S. degree from KAIST in 1985. He has got Ph. D. degree in computer science from Illinois Institute of Technology, Chicago in 1993. His major field of study is software engineering, specially software testing.

He was on the technical staff of computer research center working on the computer code standards for Korean characters and software standards for Korea Institute of Standards and Science like NIST in U.S.A. He developed an experimental Ada maintenance environment based on a syntax-directed editor for supporting program understanding. He constructed a number of CASE tools with an emphasis on integration mechanisms of tools and Object-Oriented Software Metrics. He has been a full professor at the Department of Computer Engineering, Dongguk University, Seoul, S. Korea since 1993.

Prof. Choi is a senior member of Korea Information Science Society from 1985 and a member of governing board in SIG Software Engineering, Korea Information Science Society since 1998. He was a member of editorial board, Journal of Korea Information Processing Society since 1997.