

# Task-Driven Approach and Knowledge Transfer in Practical Courses Teaching

Liangliang Kong

College of Engineering

Shanghai Second Polytechnic University

2360 Jinhai Road, Pudong District, Shanghai 201209, China

llkong@sspu.edu.cn

Lin Chen

College of Engineering

Shanghai Second Polytechnic University

2360 Jinhai Road, Pudong District, Shanghai 201209, China

chenl@sspu.edu.cn

**Abstract**—This paper analyzes the task-driven teaching and knowledge transfer at first, and then proposes a combined teaching approach based on both of them according to the characteristics of practical courses. Taking the course of computer architecture practice as an example, it explains the application of the proposed approach in teaching. Firstly, teaching content breaks down into tasks by teachers and students prepare related documents. Secondly, teachers make questions with the aim of achieving each task, and students are divided into several groups to solve the questions with the guidance of teachers. In this step, teachers may need to discuss with students or operate how to solve a problem in classroom. Finally, teachers make knowledge transfer to extend students' knowledge and enhance their interests of learning. Teachers tell students how to use the experimental data obtained before to implement and train a neural network to solve the problems of computer architecture, e.g. the estimation of performance of a target architecture or task scheduling in distributed systems. The feedback from teachers and students has shown the combined teaching approach is an effective way to improve teaching effect for practical courses.

**Keywords**—task-driven; knowledge transfer; simulator; neural networks

## I. INTRODUCTION

Nowadays, the main teaching approaches are project-based learning, task-driven, flipping classroom, etc. Since each teaching approach has its own advantages, a teacher may use several teaching approaches in a combined and flexible way in his classroom according to its teaching goal and teaching content. Teachers of traditional practical courses may focus on the ability with which students apply theoretical knowledge in practice; however, they may ignore the internalization of knowledge and knowledge transfer in their classroom. To compensate this disadvantage, this paper proposes a combined teaching approach based on task-driven teaching and knowledge transfer for practical courses. The approach can not only train the students' ability of practice but also implement the internalization and transfer of knowledge with heuristic guidance of teachers in the process of task-driven teaching. To illustrate our idea, the proposed teaching approach was applied in the teaching of computer architecture practice with the aim of exploring a more effective teaching approach which could make students become more proactive in classroom.

---

This paper has been supported by the Project of Core Course "Computer Architecture" and the Discipline of Computer Science and Technology of Shanghai Second Polytechnic University.

## II. RELATED BACKGROUND

Today, the main teaching approaches are project-based learning, task-driven, flipping classroom, etc. Project-based learning (PBL) is a methodology which aims at a whole project. Project-based learning is a student-centered pedagogy that involves a dynamic classroom approach in which it is believed that students acquire a deeper knowledge through active exploration of real-world challenges and problems [1]. Flipped classroom is an instructional strategy and a type of blended learning that reverses the traditional learning environment by delivering instructional content, often online, outside of the classroom. It moves activities, including those that may have traditionally been considered homework, into the classroom. In a flipped classroom, students watch online lectures, collaborate in online discussions, or carry out research at home and engage in concepts in the classroom with the guidance of a mentor [2]. The combined teaching approach proposed in this paper is based on task-driven teaching and knowledge transfer.

### A. Task-driven Teaching

Task-driven approach is a teaching model based on the constructivist teaching theory. In this teaching model, teachers design teaching content into one or more specific tasks, and let students master teaching contents to achieve the target of this course by completing some specific tasks. Students take initiatives in learning and teachers give guidance in this approach. Tasks as the main line, teacher as the leadership and students as the main body are the basic characters of this teaching approach. It can help students get learning initiative and study with questions, which can not only enhance student's study interest but also help the student to culture innovation ability and they can experience more pleasure[3][4].

### B. Knowledge Transfer

Knowledge transfer refers to sharing or disseminating of knowledge and providing inputs to problem solving [5]. In organizational theory, knowledge transfer is the practical problem of transferring knowledge from one part of the organization to another. Like knowledge management, knowledge transfer seeks to organize, create, capture or distribute knowledge and ensure its availability for future users. It is considered to be more than just a communication

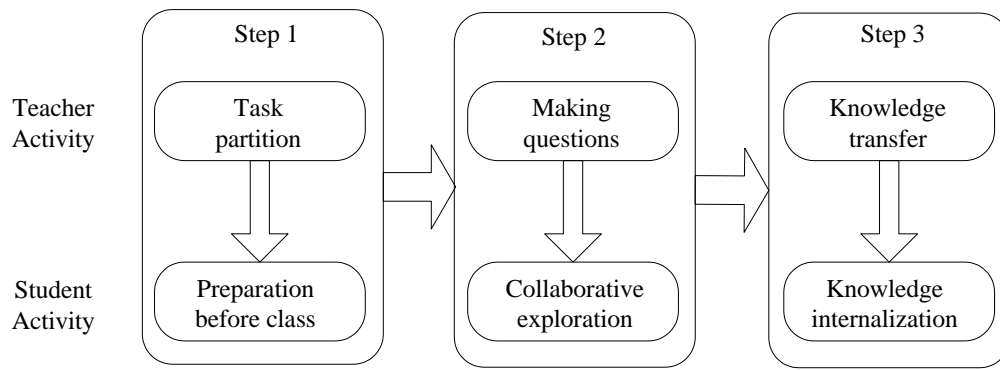


Fig. 1. The combined teaching approach based on task-driven and knowledge transfer

problem. If it were merely that, then a memorandum, an e-mail or a meeting would accomplish the knowledge transfer.

### III. SPECIFIC APPLICATION OF THE COMBINED TEACHING APPROACH IN COURSE OF COMPUTER ARCHITECTURE PRACTICE

In order to create a good teaching environment of classroom, there are several effecting factors teachers should consider, such as teaching goals, teaching content, ability of students. As a compulsory major course, the course of computer architecture practice aims at helping students to culture practical and innovation abilities and enhance skills and knowledge. To achieve the teaching goal, the combined teaching approach proposed in this paper was introduced and applied in the teaching of the course as shown in Fig.1.

#### A. Task Partition & Preparation before Class

It is important for teachers to make task partition for students to make them clear what they should do step by step. One of the teaching content of the course of computer architecture practice is to make students enable to manipulate the architecture simulator, i.e. simplescalar/arm 错误!未找到引用源。 , to obtain the function and performance metrics of the benchmark suits, i.e. Mibench. The function metrics are mainly the executed instruction counts of the benchmarks, while, the performance metrics are the execution cycles of the benchmarks, the ratio of cache hit, the ratio of transfer lookaside buffer(TLB) misses, and so on. Then, students should use these metrics to implement and train a neural network to predict performance of an application under estimation, e.g. the execution cycles. At last, teachers will take examples to show how to use the predicted results to estimate a target architecture or schedule tasks in real-time systems. According to the ability of students, the teaching content of the course is divided into several tasks as shown in TABLE I.

Before class, teachers should provide students in the course website with related documents which could be downloaded and learned by students, such as guidance book of experiments, teaching videos, source code of the simulator, cross-compilation software, source code of the benchmark suit, Matlab and its toolbox.

#### B. Making Questions & Collaborative Exploration

Teachers could extract questions from tasks divided in section III.A. Students try to solve these questions by

themselves step by step with the guidance of teachers. For easy questions, students are encouraged to solve them by themselves to culture their practical ability. For difficult questions, teachers may need to operate how to solve them in classroom on one hand and divide students into several groups with 3~4 persons in each group to discuss and solve questions in a collaborative way on the other hand. Teachers should make sure each student in a group is assigned appropriate tasks according to his interests and cognitive ability. For example, Task 3 in TABLE I can be divided into several fine-grained sub-tasks as follows: learn how to configure the simulator to match a target architecture, master how to run the simulator by manual, write scripts to run the simulator automatically, write scripts to test Mibench automatically with the simulator and record the results. Each fine-grained sub-tasks of Task 3 should be assigned to a student of a group.

During the collaborative exploration, students may face various questions or problems. In such case, teacher should inspire students to solve the problems by themselves at first, and participate the group to discuss or operate for students when necessary. During the process of helping students, the role of teacher is a guider, a monitor or consult. They just provide feedbacks to students' questions or suggestions of operation.

TABLE I. TASK PARTITION

Class hours	Tasks	Skills or knowledge students required
2	Task 1: setup and cross compiling of the architecture simulator, i.e. simplescalar/arm	Download of tools of linking and compiling, cross compiling of the simulator
3	Task 2: Cross compiling of benchmark suit, i.e. Mibench	Cross compiling and debugging of Mibench
6	Task 3: Configure and run the simulator to get the function and performance metrics of Mibench	Write the scripts to configure and run the simulator automatically and record the results of execution
4	Task 4: Implement a neural network with Matlab toolbox, and use the metrics of Mibench to train the network	Be familiar with basic principle of neural networks and application of the toolbox of Matlab
3	Task 5: Use the trained neural network to predict performance of the applications under estimation	Run the simulator to output function metrics of the application which are used as inputs of the neural network to predict performance results

### C. Knowledge Transfer & Knowledge Internalization

Knowledge transfer is the goal of learning, it refers to sharing or disseminating of knowledge and providing inputs to problem solving. In the classroom of computer architecture practice, knowledge transfer refers to usage of the function and performance metrics of Mibench which are obtained in Task 3 in TABLE I to predict performance metrics of other applications.

In this phase, teachers tell students how to implement a neural network of feed forward back propagation (FFBP) as shown in Fig.2 with Matlab toolbox at first. Then, to train the network, the function metrics of Mibench, i.e. the executed instruction counts generated by simplescalar/arm are used as input vectors of FFBP. They are classified as six types, i.e. load, store, unconditional branch, conditional branch, integer and float as shown in Fig2. The performance metrics of Mibench, i.e. execution cycles simulated by simplescalar/arm of the same benchmarks are considered as the corresponding expected outputs of FFBP. Finally, the actual output, i.e. the estimated cycles predicted by FFBP, should be compared with the expected execution cycles simulated by simplescalar/arm, and a convergence precision is obtained. If the goal of convergence precision is not met, adjust element parameters of FFBP and then continue to train the network in the next iteration. This training phase performs iteratively until the goal of convergence precision or the maximal iteration count is met. The trained FFBP network could accept the function metrics of a application under estimation, e.g. counts of instructions, as its inputs and generate the performance metrics of the application, e.g. estimated execution cycles, as its output.

For an example, 15 benchmarks of Mibench were randomly selected to test the effectiveness of the trained FFBP network. The executed instruction counts of these benchmarks simulated by simplescalar/arm in Task 3 were offered as input vectors, and then the trained FFBP network predicted execution cycles of benchmarks as its outputs. Comparing the expected execution cycles simulated by simplescalar/arm in Task 3 with the estimated ones predicted by FFBP as shown in Fig. 3, an average error of 13.8% could be concluded. Such prediction results could be accepted considering the low cost of prediction.

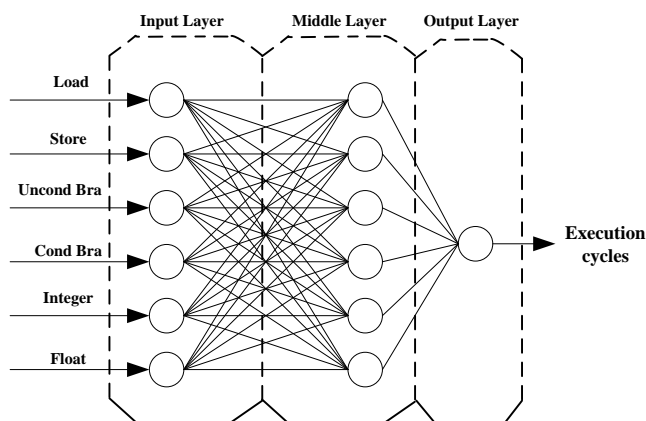


Fig. 2. The topology and usage of FFBP

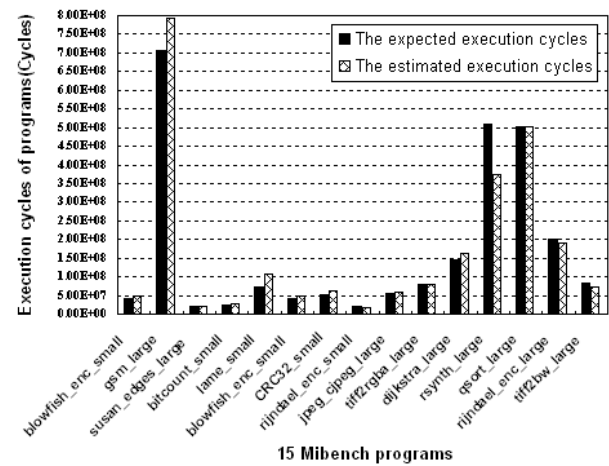


Fig. 3. Comparison of prediction results and simulation results of 15 Mibench benchmarks.

The performance metrics predicted by FFBP could be used to compare performance of different computer architectures, schedule tasks in real-time system, and so on. Teachers may take specific examples of applications of the performance metrics predicted by FFBP networks to enhance student's study interest and they can experience more pleasure. In this phase, knowledge transfer makes students enhance the comprehension of architecture simulator, be familiar with the principle of neural networks, and master the simulation method by Matlab. As a result, the knowledge and ability of students are enhanced and extended.

### IV. CONCLUSIONS

Since it is important to help students to culture innovation ability and enhance their study interest, there is a strict requirement for practical course design. This paper proposes a combined teaching approach based on task-driven teaching and knowledge transfer to meet the requirement and applies it in the course of computer architecture practice. After introducing the approach, teachers make task partition at first to guide students step by step. Then they make questions about how to complete each task and divide students into groups to discuss and solve these questions mainly by themselves. Finally, teachers make knowledge transfer to extend students' knowledge and students implement knowledge internalization through experimental experience. The feedback from teachers and students has shown it is a good way to improve teaching effect. However, it is just a try for teaching approaches of practical courses. We need to make more efforts in future to improve teaching effect more effectively.

### REFERENCES

- [1] John Dewey. Education and Experience. New York: Macmillan. 1997.
- [2] V Strauss. The flip: Turning a classroom upside down, Washington Post, 4. June, 2012.
- [3] Xin Li, Yanfei Peng, Jinguang Sun. Task-driven and cooperative-working based compiler principle teaching reform, CSEE 2011, pp 448-451.
- [4] Chuzhen Li. Task-driven approach and flipping classroom in web page design courses teaching, Fujian Computer, volume 9, 2016, pp138-139.(In Chinese)

- [5] THE KNOWLEDGE-BASED ECONOMY. ORGANISATION FOR ECONOMIC CO-OPERATION AND DEVELOPMENT. 1996. Retrieved 29 January 2017.
- [6] D Burger, Todd Austin. The SimpleScalar Tool Set, Version 4.0, Technical Report #1342. Madison: Computer Sciences Department of Wisconsin University, 2001.