Proposed Optimal Growth Pathfinding Method Based on Growth Trajectories

Niu Woyuan, Department of Computer Science and Intelligent System, Osaka Prefecture University, Sakai, Japan

Ryosuke Saga, Department of Computer Science and Intelligent System, Osaka Prefecture University, Sakai, Japan

Hiroshi Tsuji, Department of Computer Science and Intelligent System, Osaka Prefecture University, Sakai, Japan

Yukie Majima, Department of Computer Science and Intelligent System, Osaka Prefecture University, Sakai, Japan

ABSTRACT

In this study, the authors propose an optimal growth pathfinding method to support learners in effectively mastering a set of capabilities. Under the assumption of prerequisite relationships among learning objectives, the main processes of the method are as follows: (1) extracting the capability structure from growth trajectories, (2) remodeling the problem as a traveling salesman problem with restrictions among learning objectives, and (3) generating the cost matrix and obtaining the optimal growth path. In addition, a flexible approach to data standardization as a step of capability structure extraction is discussed. The proposed method is also applied to a software engineer growth dataset with 30 responders.

Keywords: Capability Space, Growth Path, Knowledge Structure, Trajectory Mining, Traveling Salesman Problem (TSP)

INTRODUCTION

To master a set of capabilities, people prefer to determine which one they have to learn first. However, making such a plan when complicated relationships exist among skills is difficult. Meanwhile, a large quantity of growth logs has been accumulated by learning support systems, such as an intelligent tutoring system. How to utilize these logs to improve learning support (learning analytics) has been the focus of considerable attention, and its development is significantly benefiting from data mining and knowledge visualization.

DOI: 10.4018/IJKSS.2015100105
As discussed in our previous research, trajectory mining on the capability space (Tsuji et al., 2013) has been used in growth log visualization and capability structure extraction (Nakamura et al. 2012) on a two-dimensional capability growth domain (called capability subspace). Growth log visualization focuses on ensuring that growth paths can be handled more easily. As such, exceptional and conventional learning paths can be reviewed more conveniently. Capability structure extraction is used to derive the potential constraints among the learning objectives (capability levels). Such constraints are usually difficult to recognize and may possibly result in ineffective growth because some growth goals are difficult to achieve with their prerequisites unreached. For example, in an English study domain, mastering the normal level of the capability “reading” is a prerequisite for the high level of the capability “listening.”

Growth trajectory is obtained by visualizing growth logs based on knowledge space theory (KST) (Doignon & Falmagne, 1985), which is usually used for knowledge acquisition. The visualized growth trajectory reflects the growth order taken by antecessors and then implies constraints. An algorithm based on the interpretive structural modeling (ISM) method (Warfield, 1982) was proposed for capability structure extraction. Although trajectory mining is theoretically feasible, it results in some problems in practical application, such as the spiral capability enhancement (SPICE) support system (Nakamura et al., 2011). Although the reachability of each learning objective (whether level A can be mastered) is identified, cost elements (time or money) are omitted. For instance, some well-known growth paths may take more time than exceptional growth paths. Although growth paths on two capability spaces are reflected clearly by visualization, growth paths on three capability growth spaces are difficult to comprehend.

In this study, a method to utilize sequential growth logs for growth path advising is considered theoretically and experimentally. We shall focus on the following topics: (1) Prerequisite time (considered as cost) between learning objectives is calculated using the proposed cost function, in that the capability structure is converted into a directed weighted graph. (2) After the first process, the optimal growth path problem is redefined as traveling salesman problem (TSP) with constraints. Then, the TSP is partly solved using the improved particle swarm optimization (PSO) algorithm (Chen et al. 2008). As the method needs to use capability structures, a solution to data standardization is used to ensure a more comprehensive structure identification. These topics are also validated in a dataset from a Taiwanese software company.

This paper is organized as follows: First, related theory and previous results, which are the basis for further discussion, will be reviewed. Then, the research problem in this study will be defined and described theoretically and illustrated using examples. The proposed optimal growth pathfinding method will be discussed in detail in connection with the existing problems. The study will be concluded with an application example to elucidate our proposed method.

**RELATED THEORY AND PREVIOUS RESULTS**

The formal concepts of capability space and capability structure are derived from the analogy of KST (Doignon & Falmagne, 1985). The basic concepts of KST and the definitions proposed by Tsuji et al. (2013) will be briefly discussed. Doignon and Falmagne (1985) proposed the KST, which provides a formal means to describe the structure of a given domain of knowledge. Knowledge state is a set of knowledge items (problem) that an individual is able to solve. Prerequisite relationships (also formalized by surmised relationships) between knowledge items restrict the set of possible knowledge states. The set of all possible knowledge states is called a knowledge space (Stahl, 2011).
An Architecture-Based Adaptation Framework for Soft Real-Time Applications
www.igi-global.com/article/architecture-based-adaptation-framework-soft/48260?camid=4v1a

A Machine Learning Based Meta-Scheduler for Multi-Core Processors
www.igi-global.com/chapter/machine-learning-based-meta-scheduler/63584?camid=4v1a