

The Effectiveness of Summer Professional Development for K-8 Computer Science Teachers

Gwen Nugent

Nebraska Center for Research on Children, Youth, Families and Schools
University of Nebraska-Lincoln, USA
gnugent@unl.edu

Keting Chen

Department of Child, Youth and Family Studies
University of Nebraska-Lincoln, USA
ke-ting.chen8985@huskers.unl.edu

Leen-Kiat Soh

Department of Computer Science and Engineering
University of Nebraska-Lincoln, USA
lksoh@cse.unl.edu

Abstract: The growing interest in offering computer science (CS) in public schools has illuminated the need for more trained K-8 educators. This paper provides initial evidence that carefully structured professional development (PD) that focuses *both* on CS skills/concepts and pedagogy can successfully impact teacher outcomes. Testing before and after the summer PD showed significant increases in teachers' knowledge of CS concepts and computational thinking, as well as confidence in their CS skills and pedagogy. The only moderating effect was for rural versus urban differences in CS confidence.

Introduction

Growing national computing workforce demands, coupled with the need for younger students to develop skills and competencies in problem solving and critical/computational thinking, has spurred introduction of computer science in K-8. A recent Gallup survey showed that two-thirds of parents want their children to learn computer science (Gallup, 2015), but data from the 2017 National Assessment of Educational Progress showed that less than half of US students in grades 4 and 8 have access to such classes (Von Zastrow, 2018). In order to offer computer science (CS) more broadly, we need K-8 educators who are trained in teaching computer science. There is also a lack of research as to what constitutes effective CS instruction and what pedagogies and instructional strategies are most appropriate to foster student learning. The research base in CS is clearly deficient in comparison to mathematics or science education, particularly in grades K-8.

To help address such deficiencies the *CS for All* Initiative was initiated by the U.S. Office of Science and Technology Policy (White House, 2016). This ongoing effort seeks to accelerate efforts to expand CS in K-12 schools and bring together federal agencies to support professional development for educators to teach CS. As part of that undertaking, the National Science Foundation released funding to support programs enabling *all* U.S. students the opportunity to participate in computer science in their schools. This paper presents strategies and research results from one of these funded projects, focusing on teacher professional development for K-8 CS.

Description of Professional Development

The professional development (PD) program consisted of two one-week summer graduate courses focusing on CS content and CS pedagogy. The summer content course was taught by a university computer science professor and dealt with fundamental CS topics (i.e., simple Input/Output, data structures, arrays, functions, search and sort) and computational thinking (CT) topics (i.e., decomposition, pattern recognition, abstraction, generalization, algorithm design, and evaluation). The course involved lectures, hands-on group activities, reflections, and homework assignments. The end-of-course project allowed teachers to pick one CS concept and CT topic and create a lesson for their targeted grade level. The second course was taught by master elementary, middle and high school CS teachers and focused on CS pedagogy and how to teach the CS concepts of loops, variables, conditionals, and functions at the elementary, middle school and high school level. High school was included in order for teachers to understand the curricular learning progression across the K-12 grade span. The instructors used hands-on activities and presentations to give teachers experiences with instructional strategies they could use in their classrooms.

Research Methodology

The study investigated two research questions: (1) What is the impact of the summer PD on teacher's a) knowledge of computer science concepts and computational thinking, b) CS self-efficacy, and c) CS attitudes? (2) Are there differences between teacher outcomes in terms of their years of teaching computer science, their knowledge of programming languages, their past CS PD experiences, and their district classification as rural or urban?

The research utilized a repeated measure, pre-post design measuring the key outcomes prior to participation in the summer PD and following the professional development.

Participants

Participants were 29 teachers, primarily from an urban district ($n = 18$) with an established K-8 CS curriculum that has been recognized nationally by CSTA for their computer science instruction. The remaining 11 were predominantly rural districts from the same state. Most participants in the overall sample were female (76%). Teachers average age was 45, with 17 years of teaching experience, and 5 years in teaching computer science. 76% had a master's degree. The average percent of free and reduced lunch per school was 48%, ranging from 4% to 91%. Rural – urban district comparisons showed similar demographics between the two groups with the exception that the urban district had more male CS teachers and an overall higher free and reduced lunch percentage (51%) than the rural districts (39%).

Instruments

Teacher *knowledge* of computer science was measured by two previously validated instruments used in beginning undergraduate CS courses. One instrument focused on CS concepts such as selection statements, functions, and sorting (Shell & Soh, 2013); the second focused on computational thinking (CTCAST: Peteranetz, Morrow, & Soh, 2020).

Computer science *self-efficacy* was determined through a project-developed 31-item confidence instrument measuring two constructs: a) confidence in teaching computer science (16 items; e.g., *I can assist all students who are having trouble mastering specific programming/computer science skills*) and b) confidence in their CS skills (6 items; e.g., *I can design and iteratively develop/refine CS programs*). Items were rated on a 0–100% confidence scale and were developed to align with objectives of each of the summer courses.

The *attitudinal* items used a Likert scale (1: strongly disagree, 2: disagree, 3: neutral, 4: agree, 5: strongly agree) to measure personal interest in CS (e.g., *I find the challenge of solving computer science problems motivating*) and the perceived value of CS (e.g., *Reasoning skills used to understand CS can be helpful to me in my everyday life*). The teacher instrument was developed by adapting the *Computing Attitudes Survey* (Dorn & Tew, 2015), which was validated with CS undergraduates.

Results

There were significant pre-post gains in teachers' knowledge of both computer science concepts and computational thinking (Table 1). There was a greater increase in their knowledge of CS concepts than computational thinking (Figure 1), explained partly by the very low concept pre scores (23%). The PD introduced teachers to higher-level concepts than they were used to teaching in K-8. Of note is that the concept test was designed to separate high performers from low performers, so instead of a C-average being around 70%-80%, the average test scores were intended to be around 50%. The teacher's higher scores for computational thinking were expected given their CS teaching experience.

Outcome	Pre Mean/SD	Post Mean/SD	N	t-result
CS Knowledge				
Concepts	3.83 (29%)/2.36	6.43 (50%)	29	$t(28) = 5.78, p < .001$
Computational thinking	9.52 (53%)/3.62	11.77(65%)	29	$t(28) = 3.58, p < .01$
CS Self-efficacy	73.51/21.69	83.40/11.26	24	
CS Content	62.15/26.15	71.89/23.62		$t(23) = 2.96, p < .01$
CS Pedagogy	78.88/17.01	88.34/9.35		$t(23) = 4.36, p < .001$
CS Attitudes	4.52/.43	4.60/.32	24	
Personal interest	4.51/.51	4.51/.42		$t(23) = \text{not significant}$
Perceived Value	4.54/.45	4.71/.34		$t(23) = 1.84, p = .08$

Table 1. Participant descriptives and results

There was also a significant pre-post gain in teacher's *confidence* in their CS skills and pedagogy. The teachers came into the professional development (pre scores) with higher confidence in pedagogy than content. This may again be due to their lack of familiarity with the higher level concepts typically taught in an undergraduate CS class.

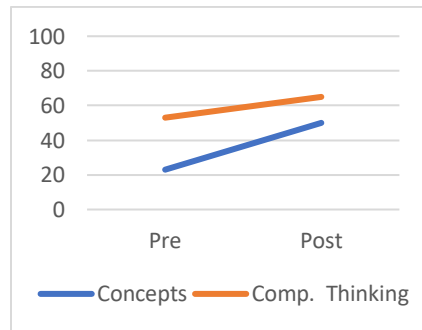


Figure 1. Pre-post Knowledge

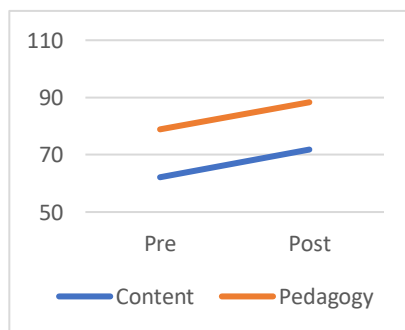


Figure 2. Pre-post Confidence

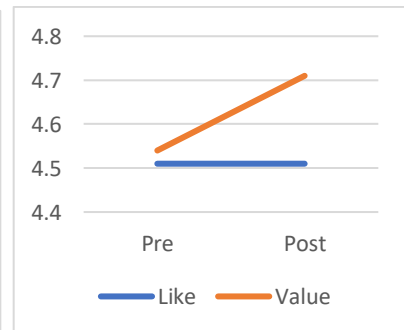


Figure 3. CS Pre-post Attitude

In contrast to the significant results for knowledge and confidence, the teachers' *attitudes* towards computer science did not show increases. Figure 3 shows that their personal interest in CS remained stable, while their perception of the value of computer science increased, although not significantly ($p = .08$). The teachers had very high ratings coming into the PD, which resulted in a ceiling effect that limited significant increases.

Statistical analyses examining possible moderating effects (research question 2) showed no moderation in terms of years of teaching experience, knowledge of programming languages, or past CS PD experiences. However, split plot analyses between urban and rural teachers showed one significant effect: there was a significant interaction between time (pre-post) and rural – urban designation for the teacher self-efficacy outcome (Wilks' $\Lambda = .77, F(1,22) = 6.77, p < .05$). While the urban district had higher pre confidence scores, the rural teachers had a steeper increase in scores from pre to post.

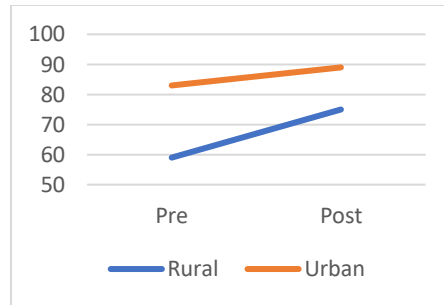


Figure 4. Differences in rural and urban teachers pre-post confidence

Discussion

A comprehensive survey of pre-college computer science education conducted by Google concluded that “teacher development is a key factor in the success of CSEd” (Bilkstein, 2018, p. 10). The report goes on to say that traditional college -level practices should not be automatically used in K-8 environments and that the field has not yet achieved a solid body of K-8 CSEd research. This study provides initial evidence that carefully structured PD that focuses *both* on CS skills/concepts and pedagogy can successfully increase teachers’ knowledge and confidence in teaching CS. Teachers need more than knowledge *about* computer science, they need *skills and confidence* that they can successfully deliver CS instruction. Providing hands-on, concepts-based activities that can be utilized in the classroom appears to bolster teacher confidence. Since CS represents a new subject in public schools and requires new pedagogical approaches, research studies that focus on elements of effective PD are critical.

The fact that there was only one moderating effect may be explained by the relatively small sample ($n = 29$), which meant that examining differences between subgroups was underpowered. The significant effect showed a difference in pre-post confidence between rural and urban teachers. This result may be due to greater district-level support for urban teachers, including established curriculum and learning progressions, local workshops, and professional learning communities. In addition, some of the grade 6-8 urban schools have multiple CS teachers, providing collaboration opportunities. This rural-urban difference illustrates the challenges facing rural districts in terms of access to high-quality CS courses and teacher PD (CSTA, 2018).

References

- Blikstein, P. (2018). Pre-college computer science education: a survey of the field. Mountain View, CA: Google LLC. <https://services.google.com/fh/files/misc/pre-college-computer-science-education-report.pdf>.
- Dorn, B & Tew, A. (2015). Empirical validation and application of the computing attitudes survey. *Computer Science Education*, 25, 1-36.
- Gallup (2015). Searching for Computer Science: Access and Barriers in U.S. K-12 Education. https://services.google.com/fh/files/misc/searching-for-computer-science_report.pdf.
- Peteranetz, P., Morrow, P., & Soh, L-K. (2020, March). *Development validation of the computational thinking concepts and skills test*. Paper presented at SIGCSE 2020, Portland, OR.
- Shell, D & Soh, L-K. (2013). Profiles of motivated self-regulation in college computer science courses: Differences in major versus required non-major courses. *Journal of Science Education and Technology*, 22, 899-913.
- Von Zastrow, C. (2018, September 11). Do K-8 student have opportunities to learn computer science? *Ed Note policy blog*. <https://ednote.ecs.org/do-k-8-students-have-opportunities-to-learn-computer-science/>.

White House. (2016). FACT SHEET/ A Year of Action Supporting CS for All. <https://obamawhitehouse.archives.gov/the-press-office/2016/12/05/fact-sheet-year-action-supporting-computer-science-all>.

CSTA. (2018). State of Computer Science Education: Policy and Implementation. https://code.org/files/2018_state_of_cs.pdf.

Acknowledgements

This research was supported by a grant from the National Science Foundation award #1837476.