Why Discipline Matters in Computing Education Scholarship

JOSH TENENBERG
University of Washington
and
ROBERT McCARTNEY
University of Connecticut

Scholarly communities in computing determine how to produce and validate knowledge within their domains of focus. Much of this knowledge can remain tacit because of shared ways of becoming a disciplinary scholar within any particular area of computing. But such tacitness presents challenges for computing education scholarship, since knowledge about how to create and validate claims about human thinking and learning is rarely a part of the training for computing educators. The editors-in-chief of TOCE have responded to these challenges by making explicit the epistemic practices associated with the publication of its manuscripts, by encouraging a range of borrowing from other disciplines, and by instituting feedback loops at all levels of review.

Categories and Subject Descriptors: K.3.2 [**Computers and Education**]: Computer and Information Science Education—*Curriculum; Computer Science Education*

General Terms: Human Factors

Additional Key Words and Phrases: Computing education, scholarship of teaching and learning, epistemic culture, community of practice

ACM Reference Format:
Tenenberg, J. and McCartney, R. 2010. Why discipline matters in computing education scholarship. ACM Trans. Comput. Educ. 9, 4, Article 18 (January 2010), 7 pages.
DOI = 10.1145.1656255.1656256. http://doi.acm.org/10.1145.1656255.1656256.

In this editorial, we focus on the unique challenges associated with engaging in and publishing computing education scholarship. These challenges arise because of the ways in which knowledge communities create and validate knowledge as well as the fact that computing education scholarship relies upon theory and method borrowings from outside computing. In this editorial,

Authors' addresses: J. Tenenberg, Institute of Technology, University of Washington, Tacoma, 1900 Commerce St., Tacoma, WA 98402; email: jtenenbg@u.washington.edu; R. McCartney, Department of Computer Science and Engineering, University of Connecticut, Storrs, CT 06269-2155; email: robert@engr.uconn.edu.

we highlight these challenges and discuss how we have responded in policy and practice in determining how manuscripts are chosen for publication in TOCE.

1. DISCIPLINARY KNOWLEDGE

Scholarly disciplines have distinctly different ways of working with knowledge. How a historian goes about constructing and working with knowledge differs from the knowledge practices of an anthropologist, both of which differ from those of the economist [Lamont 2009]. This is what Knorr-Cetina calls the *epistemic culture* of a scientific community: "those sets of practices, arrangements and mechanisms bound together by necessity, affinity and historical coincidence which, in a given area of professional expertise, make up how we know what we know. Epistemic cultures are cultures of creating and warranting knowledge" [Knorr Cetina 2007, p363]. This is as true of the disciplines that comprise computing (e.g., information systems, computer engineering, computer science) as it is of any other discipline.

Evaluating what constitutes good or "publishable" research requires answering a number of questions, which include:

—does the research address an important issue?
—is there a literature upon which this work builds?
—are the methods appropriate to the task at hand? applied with care?
—is there data? how collected? from whom or what? how much? how analyzed?
—does the article follow standard forms of reporting?

Only by working within disciplinary areas of computing (or any other discipline), can individuals answers these question. Yet much of the knowledge needed to determine these answer is taken for granted, what Polanyi [1966] termed *tacit* knowledge. In general, the tacitness of knowledge that underlies discourse within a disciplinary community is not problematic, since members of that community have come to internalize the shared paradigmatic knowledge through their socialization. Thus, disciplines matter by providing the mechanisms for knowledge creation and validation and the basis for evaluating research quality.

2. INQUIRY INTO DISCIPLINARY TEACHING AND LEARNING

If experts can evaluate the quality of scholarship concerning in an area of computing such as databases, what about scholarship concerning the teaching and learning of databases? Educational research indicates the importance of deep disciplinary knowledge in teaching [Bransford et al. 2000]; databases cannot effectively be taught without considerable subject knowledge. In addition, skilled teachers possess considerable *pedagogical content knowledge*, what Shulman defines as "the most useful forms of representation of those ideas, the most powerful analogies, illustrations, examples, explanations, and demonstrations—in a word, the ways of representing and formulating the subject that make it comprehensible to others" [Shulman 1986, p. 9]. Clearly, disciplines and disciplinary knowledge matter when it comes to teaching and learning within the discipline.

But disciplinary knowledge alone is insufficient for many kinds of inquiry into teaching and learning in computing, especially inquiry that goes beyond the Marco Polo ("been there, done that") and tools articles that characterize so much of the educational work that is published in computing [Valentine 2004]. This is because *making and verifying claims about human learning differ fundamentally from designing, building, and using computational artifacts*. To study human learning in computing, we need not only disciplinary knowledge (to understand what the important questions are, to interpret student behavior and artifacts), but we need knowledge about the teaching and learning processes: how students learn, how learning and teaching interact, and how the effectiveness of these might be evaluated. Researchers in the social and behavioral sciences in particular have already developed epistemic cultures of considerable subtlety and depth that can provide insight into many of the questions about teaching and learning that computing educators might ask.

3. IMPLICATIONS FOR TOCE

The challenge then, is that although we might share knowledge about what is publishable in the computing disciplines, we have no such shared knowledge about doing so in computing education. We each bring in our own assumptions about "publishability," since most computing educators have not been socialized into a practice community in the social and behavioral sciences. And most of these assumptions are tacit, taken for granted, and unexamined. While tacitness is not problematic within the discipline, where community members have common and shared knowledge, tacitness is problematic in building communal knowledge about what constitutes excellent scholarship in computing education.

What are the implications for TOCE?

3.1 Explicitness

If tacitness is the problem, then explicitness is one of the ways of addressing it. As editors-in-chief of TOCE, we have promoted explicitness at all levels. First, we have articulated our vision for quality research in computing education in our editorials (both in TOCE and its predecessor JERIC) [Tenenberg and McCartney 2009; Tenenberg and McCartney 2007b; 2008a; 2008b; Tenenberg and McCartney 2008; 2007; Tenenberg and McCartney 2007a].

Second, we have operationalized this vision in explicit review criteria that all papers must meet, as listed on the TOCE Web site: toce.acm.org. This criteria was informed by meta-studies on the characteristics of excellence in both disciplinary and educational research across a range of disciplines [Glassick et al. 1997], and as represented in a number of journals with similar missions (such as *Computer Science Education* and the *Journal of Engineering Education*).

Third, we hold reviewers and associate editors accountable. They must use this criteria in making their decisions, and these decisions must come with discussions on the reasons why particular recommendations and choices have been made. And our associate editors are excellent scholars and scientists who hold us accountable to the same standards.

3.2 Feedback Loops

Building a shared epistemic culture requires that discussion not simply be uni-directional: from authors to reviewers to associate editors, to editors, and then back to authors. Rather, the decisions that are made at each level need to be made visible to those at other levels. Associate editors should know whether editors have accepted their preliminary decisions and why. Reviewers should see how their decisions impacted subsequent drafts. Without such self-corrective mechanisms, the members of the computing education communities cannot cannot otherwise develop paradigmatic ways of thinking about disciplinary education scholarship.

3.3 Methodological Ecumenicalism

The methods used in particular areas of disciplinary inquiry are social and historical products. They have arisen not only because of their “fitness” to the problems that the disciplinary community has set for itself. But they have arisen through a social and political process within the community, as different subgroups contest what it means to construct and validate knowledge within that community. But in something as broad as computing education, where not only is there great variety in the object of investigation (grouped into ten areas by Fincher and Petre [2004], such as “student understanding, teaching methods, and recruitment and retention of students), there is an immense variety of methods of investigation; our colleagues in other disciplines have not been idle. If we couple this recognition with the fact that computing education is in its early development as a discipline (and far from paradigmatic), it argues for methodological ecumenicalism. We have far more to gain by profligate method borrowing than we do by prematurely closing off methods of inquiry, even if that presents challenges to readers, reviewers, and editors. We thus seek out reviewers to ensure that there is both disciplinary expertise and expertise in pedagogical inquiry, asking colleagues from other disciplines to review as needed.

3.4 Co-Editorship

Most ACM journals and transactions have a single editor-in-chief who makes the final decisions on what to publish. As co-editors-in-chief, we decide by reaching agreement: we discuss each manuscript that is submitted and closely track its progress as it moves through the review cycle, and we discuss the decisions until we agree. Reaching such agreement requires that we make our reasoning visible to one another, ensuring that we have a consensus decision on any manuscript that is accepted for publication.

In short, those of us engaging in computing education scholarship are under obligation to go beyond the bounds of our own disciplinary practices. This is not an easy task, especially given the demands of teaching and of carrying out disciplinary research. As individuals, we are limited in the extent to which we can borrow (let alone master) the methods and theories from other disciplines. "Only as an ideal can I imagine a researcher so talented that any appropriate method can be brought to bear on a problem encountered" states the educational anthropologist Wolcott [1992, p. 8]. He adds: "The most skilled and experienced fieldworkers probably have only one or two sharply-honed research skills upon which to draw." Colleagues from other disciplinary areas can be recruited for their expertise, though this will mean our acquiring at least *interactional* expertise of those disciplines: "[t]his is expertise in the *language* of a specialism in the absence of expertise in its *practice*" [Collins and Evans 2007, p. 28] (emphasis in original).

There are as well emerging communities of practice in inter-disciplinary research in computing education, and computing educators can become "legitimate participants" Lave and Wenger [1991] in these communities. Avenues of access include the SIGCSE/ITiCSE working groups, and the SIGCSE International Computing Education Research (ICER) workshops. And finally, TOCE plays an important role by publishing the highest quality scholarship undertaken in computing education for those doing scholarship that requires both deep disciplinary knowledge as well as disciplinary knowledge from outside their discipline.

In all of these ways discipline matters.

4. IN THIS ISSUE

Each of the articles in this issue exhibit considerable expertise within the computing disciplines to address problems of teaching. In addition, each borrows methods from other disciplines, either to frame the design or study, to implement the teaching innovation, and/or to evaluate effectiveness.

Brusilovsky and his colleagues provide a discussion of their integration of a number of electronic tools for facilitating student learning of SQL in "Learning SQL Programming with Interactive Tools: From Integration to Personalization." Developing their suite of integrated tools required considerable disciplinary expertise tailored to the specific needs of their students. Their evaluation involved instrumenting the tools so as to provide statistics of use for more than two-hundred students in both undergraduate and graduate courses, along with statistical analysis typical of large-scale program evaluations in the education and social sciences. In addition, the authors developed and analyzed a questionnaire to probe student attitudes about their use of these tools, a standard methodology borrowed from the social sciences.

Yuan describes the development of visualizations for teaching specific concepts in computer security in "Visualization Tools for Teaching Computer Security." As with the other articles in this issue, the teaching intervention exploits the author's deep understanding of the subject matter as well as his understanding of how to teach this subject. The contribution of these tools to learning was evaluated through a pre- and post-test of subject matter, a design characteristic of research in education and psychology. In addition, a closed-form questionnaire designed to measure student attitudes toward their use of the tools was also administered, borrowing from social-psychological studies.

In "An Agile Constructionist Mentoring Methodology for Software Projects in the High School," Meerbaum-Salant and Hazzan address a challenging problem that arises from teaching practice: how should high school teachers mentor the software development projects of their students? In addressing this topic, the authors provide a "mentoring methodology" that gives explicit guidance to teachers who might otherwise struggle with the complexity associated with mentoring student project work. What is noteworthy about this article is that it combines disciplinary expertise about software development methodology with theories from constructionist learning and studies of teacher knowledge, both borrowed from educational research. In addition, the researchers developed their methodology via an iterative and interactive process involving dozens of high school teachers and their students who were observed and interviewed over a considerable length of time. The evaluations are mixed-method, using a number of data collection and analysis methods characteristic of evaluation research in education and the social sciences.

REFERENCES

BRANSFORD, J., BROWN, A., AND COCKING, R., Eds. 2000. *How People Learn*. National Academy Press.

COLLINS, H. AND EVANS, R. 2007. *Rethinking Expertise*. The University of Chicago Press.

FINCHER, S. AND PETRE, M. 2004. Mapping the territory. In *Computer Science Education Research*, S. Fincher and M. Petre Eds. Taylor and Francis, Chapter 1, 1–8.

GLASSICK, C., HUBER, M., AND MAEROFF, G. 1997. *Scholarship Assessed: Evaluation of the Professoriate*. Jossey Bass Publishers, San Francisco, CA.

KNORR CETINA, K. 2007. Culture in global knowledge societies: Knowledge cultures and epistemic cultures. *Interdisc. Sci. Rev. 32*, 4, 361–375.

LAMONT, M. 2009. *How Professors Think: Inside the Curious World of Academic Judgment*. Harvard University Press.

LAVE, J. AND WENGER, E. 1991. *Situated Learning: Legitimate Peripheral Participation*. Cambridge University Press.

MCCARTNEY, R. AND TENENBERG, J. 2007a. From the editors: A new vision for JERIC. *J. Educ. Res. Comput. 7*, 1.

MCCARTNEY, R. AND TENENBERG, J. 2007b. Why evidence? *J. Educ. Res. Comput. 7*, 2.

MCCARTNEY, R. AND TENENBERG, J. 2008a. From conference to journal. *J. Educ. Res. Comput. 7*, 1.

MCCARTNEY, R. AND TENENBERG, J. 2008b. Making it real. *J. Educ. Res. Comput. 7*, 3.

POLANYI, M. 1966. *The Tacit Dimension*. Doubleday.

SHULMAN, L. 1986. Those who understand: Knowledge growth in teaching. *Educ. Research. 15*, 2, 4–14.

TENENBERG, J. AND MCCARTNEY, R. 2007. Linking questions and evidence. *J. Educ. Resour. Comput. 7*, 4.

TENENBERG, J. AND MCCARTNEY, R. 2008. Grounding the scholarship of teaching and learning in practice. *J. Educ. Resour. Comput. 8*, 2.

TENENBERG, J. AND MCCARTNEY, R. 2009. Introducing the ACM transactions on computing education. *Trans. Comput. Educ. 9*, 1.

VALENTINE, D. W. 2004. CS educational research: A meta-analysis of SIGCSE technical symposium proceedings. In *Proceedings of the 35th Technical Symposium on Computer Science Education (SIGCSE'04)*. 255–259.

WOLCOTT, H. 1992. What qualitative research has revealed about education's researchers. In *Proceedings of the American Educational Research Association Symposium (AERA'92)*.

Received November 2009; accepted November 2009.