Teaching Journalism and Mass Communication, Vol. 6, no. 1, pp. 77-91, Winter 2017
http://www.aejmc.us/spig/journal

Collaborating Across Boundaries to Engage Journalism Students in Computational Thinking

Kim Pearson
Department of Journalism and Professional Writing
The College of New Jersey

Monisha Pulimood
Department of Computer Science
The College of New Jersey

Diane Bates
Department of Sociology & Anthropology
The College of New Jersey

Journalism educators seek ways to create a positive environment for learning computational journalism. This paper describes a multi-semester collaboration between undergraduate journalism and computer science students. Data indicate that such collaborations can strengthen journalism students' confidence in their ability to employ computing tools and methods. However, journalism students did not show as much positive change as did students in computer science and other majors. Future research will focus on student preparation for such collaborations. This research contributes to the search for teaching and curriculum design strategies for integrating computational thinking into the journalism curriculum.

The push to enhance journalism students' computing skills presents a welter of curricular and pedagogical challenges and opportunities. While growing attention and debate has been devoted to the teaching of particular skills such as scripting and programming, less attention has been paid to more fundamental questions, such as, what does it mean for journalism students to think computationally? What kind of scaffolding do journalism students need in order to understand when and how to apply computing tools and processes in their work? How might they become motivated to master new computing skills, and how will they gain the confidence to persist when the effort proves challenging? And finally, since computational journalism at the professional level is a multidisciplinary, collaborative effort, how and where do journalism students learn these collaboration skills?

The competencies and problem-solving skills that we are seeking to inculcate have come to be known as computational thinking. While there is as yet no uniform definition, there is general agreement that *computational thinking* includes a broad range of mental tools and concepts from computer science that help people solve

Keywords: collaboration, community-engaged journalism, computational journalism, computational thinking

Correspondence: Kim Pearson, The College of New Jersey, kpearson@tcnj.edu

problems, design systems, and engage computers to assist in automating a wide range of intellectual processes. Computational thinking can be regarded as a group phenomenon as well as an individual one and this facility can assist specialists in other disciplines to more effectively adopt, use, and develop computational tools (National Research Council, 2010, p. 27).

These questions dovetail with a national effort to broaden and deepen the ranks of computing professionals, and to enhance the collaboration and communication skills of computer science students. In the last decade, formal research collaborations have emerged between journalism and computer science educators aimed at enhancing computer science instruction while broadening participation in computing through computational journalism. This paper describes CABECT (Collaborating Across Boundaries to Engage Undergraduates in Computational Thinking), a research project rooted in the belief that multidisciplinary computing collaborations at the undergraduate level and directed at real community problems will boost computational thinking, improve students' knowledge of the computing tools and processes relevant to their profession, and whet their appetites to learn more.

Specifically, we describe a multi-semester collaboration between two faculty members - one in computer science and the other in journalism and interactive multimedia, and their respective classes to build a software system to provide comprehensive, accessible information about underutilized land in an East Coast American city. Student feedback and assessment data support the hypothesis that such collaborations engage students more deeply in computational thinking. However, journalism students did not show as much positive change as did students in computer science, interactive multimedia and other majors. The data here raise questions about the best curricular preparation for such collaborations that present opportunities for future research.

Literature Review

The transformative impact of computer science on journalism practices has been widely noted and remains an object of continuous study. Royal (2015) argues that it has been so profound that the industry itself is now a technology industry. Gynnild (2014) characterizes these emergent practices as "computational exploration in journalism" which "typically involves the journalistic co-creation of quantitative news projects that transcend geographical, disciplinary, and linguistic boundaries" (p. 713). Flew, Spurgeon, Daniel, & Swift (2012) note that bringing journalists and computer science together opens up the possibility for new computing tools for mining data, contextualizing information, and engaging citizens both as news consumers and as citizen journalists.

Indeed, Coddington (2014) contends that one of the things that distinguishes computational journalism from data journalism and its antecedent, computer-assisted reporting, is that computational journalism is designed to leverage the efforts and knowledge of interests beyond the newsroom, whether they be subject-matter experts, concerned citizens, or amateur sleuths.

If we accept this typology, journalism educators are presented with the question: are we preparing students to be computer-assisted reporters, data journalists or computational journalists? Scholars and industry professionals argue that the profession is moving in favor of the latter two designations. If that is the case, how do we ensure that students are conversant with both data journalism and computational journalism, particularly in undergraduate programs?

Stray (2013) contends that computational journalists create tools that data journalists use. In other words, computational journalists have a strong background in computer science, including "standard algorithms and linear algebra" (para. 1) and some programming skill. Bradshaw (2012a) argues that all journalists should, at minimum, know how to write web scrapers—a skill one can learn without becoming a full-blown programmer. Industry professionals and academics speaking at a 2013 AEJMC panel voiced strong support for the proposition that all journalism students should be programmers (Hernandez, 2011). Royal (2013) advocates overhauling the curriculum to center on the history, culture and impact of digital media on the editorial, business, social, legal and ethical aspects of the news industry. This shift would include the ability to concentrate on multimedia, social media or programming for journalism.

At the very least, then, undergraduate programs should ensure that students are prepared to become data journalists, and have a gateway into becoming computational journalists while acquiring the other requisite skills and knowledge in storytelling, reporting, law and ethics. Indeed, this is consistent with the national consensus that the ability to think computationally is a fundamental literacy skill on a par with reading and numeracy (Wing, 2011). However, this proposition presents a variety of challenges. Royal (2014) acknowledges one challenge, which is a shortage of faculty qualified to teach these skills. Others ask how to incorporate these skills into an already crowded curriculum.

Equally challenging, though, is the pipeline problem: there is a national shortage of students who are prepared for and inclined toward computer-science education. Jane Margolis' 2008 book, *Stuck in the Shallow End,* persuasively argued that particularly in schools with high concentrations of poor students and students of color, the dearth of computing teachers and curricula perpetuates existing race and class divides.[2] Innumeracy among journalists is another obstacle - both professionals and students frequently admit to being turned off or intimidated by numbers, despite numerous efforts within the industry to help them improve their skills and gain greater confidence (Wihbey, 2015). Journalism faculty also see evidence of a tech skills gap among their students of color. Boston University professor of practice Michelle Johnson (2012) has noted, "I'm seeing that many of the students of color lack experience with the tools and technologies that will be fundamental to journalism innovation going forward" (para. 12).

Computer science educators, advocacy groups and policymakers have mounted a substantial effort to broaden and diversify the ranks of both computer science teachers and students through such initiatives as the National Science Foundation's STEM-C partnerships programs. A central part of that initiative is to encourage high schools to offer introductory and advanced computer science courses leading to an Advanced Placement examination. While increasing numbers of students are taking these courses, their numbers are still relatively small and participation by African American and Latino students lags behind. Particularly concerning is that African American and Latino students have lower pass rates on the CS AP exam than their white counterparts (Guzdial, 2014; Ericson, 2014a; Ericson, 2014b).

Many of the computing education initiatives created under the umbrella of STEM-C and other NSF-funded programs involve interdisciplinary computing collaborations. However, one is hard-pressed to find research projects involving computer science education and journalism in NSF's database (see http://search.nsf.gov). Of course, journalism educators and industry leaders are engaged in computing instruction, sometimes in the context of formal research studies, but often not (cf. McAdams, 2012; Bradshaw, 2012b).

Computing education research in journalism may make important contributions to the broader effort to understand how to make computational thinking accessible across the curriculum. Cindy Royal has argued that each discipline has its particular computing needs, and may require tailored approaches to computing instruction (Royal, 2015). She argues further that embedding computing education into the journalism curriculum not only serves industry needs, it serves the larger goal of diversifying the computing workforce, since nearly two-thirds of all journalism majors are female (Royal, 2012). Royal believes that journalism students who are programming novices might be more receptive to computing instruction in the supportive environment of the journalism classroom, as opposed to the computer science classroom. She has been building a collection of online tutorials and sample code tailored to communication students: Code, Actually (http://codeactually.com).

Royal's perspective is complemented by that of computer science educator Ursula Wolz, who observes that the process of doing journalism parallels aspects of the process of developing software. Journalists delineate a story to be covered while computer scientists define a problem to be solved. Both journalists and computer scientists make strategic decisions about the best way to accomplish tasks within defined constraints. Both groups are concerned with the validity and reliability of the data they collect and the methods they employ. Both groups produce artifacts tailored to the requirements of specific technologies and end users. Wolz argues that the student newsroom is a rich "non-didactic" learning environment for collaborative problem-solving (National Research Council, 2010, p. 22).

Background on the Collaboration

The principal investigators on this project are a computer science professor and a journalism professor who each had industry and academic experience that buttressed their philosophical inclinations toward project-focused, inquiry-based collaborative learning. Both the investigators and the evaluator who would eventually join them are professors at a primarily undergraduate institution on the east coast of the United States, near the capital city of the state.

Journalism is offered under the formal name Journalism and Professional Writing (JPW). Many classes are cross-listed with the college's interactive multimedia major (IMM), which combines writing, digital media and interactive computing. There are about 100 JPW majors, and additional students pursuing minors either in journalism or professional writing. There are two required journalism courses with explicit computing instruction. An introductory-level course, Writing for Interactive Multimedia, introduces students to html, css and usability. An advanced course, Data Journalism (formerly Computer-Assisted Reporting), introduces structured data and algorithms via spreadsheets. New elective courses such as Multimedia Journalism and Health and Environmental Reporting introduce various digital tools and techniques, such web scraping, but there is no explicit instruction in programming. A number of journalism students pursue the interactive multimedia minor, which does include explicit programming instruction in at least two of its required introductory courses, as well as a number of intermediate and advanced classes.

As is true of journalism educators everywhere, the faculty of our college has pondered the best ways to integrate computing and multimedia into curriculum and intellectual culture of our learning community. We have pursued the creation of collaborative learning environments since 1990, by partnering with classes in graphic design and television production to create multimedia content. By 1996, we had moved our collaboration online with the creation of an online newsmagazine.

The authors first began collaborating in 2006, when the computer science professor's summer research students built a content management system for the online magazine that the journalism professor and her students had been operating since 1996. The next year, we joined forces with another computer science professor to create a demonstration project (National Science Foundation Award No. 0739173) based on the hypothesis that we might broaden participation in computing by giving young people an opportunity to create multimedia and interactive stories about their own communities.

The computer science professor then collaborated with two other journalism professors to create a database manager for organizing information about the disposition of gun crimes in a major city's municipal courts. This investigation ended up contributing to an award-winning newspaper series that prompted a state takeover and reform initiative (Pulimood, Shaw, & Lounsberry, 2011). Student feedback from this project confirmed conclusions from previous research supporting the value of having students collaborate across disciplines on computing solutions to real-world problems (Pulimood & Wolz, 2008).

We conceived the CABECT project as a way of having computer science (CS) students collaborate with non-CS students and a community partner to create technology-based solutions to real-world problems. Our hypothesis is that:

> To increase motivation toward, and interest in, computing careers, undergraduate students must be immersed in multidisciplinary collaborative experiences where they are creators of computational solutions, and where they internalize the relevance of and interconnectedness between classroom learning and the community they live in.

As proof of concept, we proposed to have students in our respective classes develop a software system that made information on local brownfields easier to access and interpret. (A brownfield is an underutilized land parcel that may be contaminated.) The difficulty of obtaining accurate, comprehensive information about soil and water contamination on specific land parcels is a significant obstacle to developing affordable housing and community gardens in high-poverty areas where the needs are acute. Funding for the project was obtained from the National Science Foundation's TUES (Transforming Undergraduate Education in STEM) program (Award #1141170).

The overarching goal for the project is to formalize a model for courses that collaborate with a community partner across disciplinary boundaries. Achieving this goal required us to:

- Develop a framework for collaborating in courses across disciplinary boundaries.
- Develop a framework for collaborating in a meaningful way with a community partner.

Our **pedagogical goal** has been to create a model experiential learning environment through which to immerse both computer science and non-computer science majors in computational thinking. Our **research** goal is to study the learning environment we create, to articulate both the processes and products that manifest creative activities and problem-solving in the community. Through systematic observation and recordkeeping, we have developed quantitative and qualitative instruments that ultimately will contribute to the knowledge base of STEM workforce preparedness in computing as they are applied to creative environments beyond our institution. We envision the outcomes of the collaboration for the journalism students as follows.

- Students can describe similarities between the process of doing journalism and creating software.
- Students recognize the need for computational thinking in their own discipline.
- Positive impact on the timeliness, amplification effects and rhetorical velocity of student journalism.

The Collaboration

The collaboration has just completed its fifth semester. Table 1 includes a list of the classes involved in the collaboration each semester.

Each class runs separately with its own deliverables, but major assignments were structured around the goal of the collaboration, which has been to create a software system with mobile and social extensions that provides accurate, accessible and comprehensive information on whether a particular site is polluted. This is a complex problem, so each semester, students identified specific components of the system that they will develop or enhance.

Table 1
Classes Involved CABECT

Semester	Computer Science Class	Journalism Class
Spring, 2013	Software Engineering	Blogging and Social Media (cross-listed with IMM)
Fall, 2013	Database Systems	Health and Environmental Journalism
	Software Engineering	News Games (cross-listed with IMM)
Spring, 2014	Software Engineering	Future of the News
Fall, 2014	Software Engineering	Health and Environmental Journalism
Spring, 2015	Software Engineering	Blogging and Social Media

The students met jointly four times in the first semester. In the initial meeting, they received a presentation from the executive director and a construction manager from the local Habitat for Humanity chapter explaining how the lack of accurate pollution information complicates their efforts to provide affordable housing. (It should be noted that Habitat's position is in the collaboration was that of an expert source and an example of a community agency with an information need. It was impressed upon the students that as journalists, we were not in the business of advocating for any interest group.)

The students brainstormed ideas for software modules that would help address the problem faced by Habitat and other community stakeholders needing current, reliable, accessible environmental information. The journalism students were responsible for identifying credible data sources, and the software engineering students were responsible for the technical design and programming.

At a subsequent joint meeting, they received a presentation from an environmental policy expert who answered questions about scientific and regulatory issues. They also took a field trip to the neighborhood that Habitat is working to redevelop. The small groups were charged with the responsibility of meeting outside of class to continue working on the project. At the end of the semester, the students presented their software modules to Habitat and other interested stakeholders.

Each class also had deliverables that were not necessarily related to the collaboration. They created blogs that included at least one video segment and a Google Fusion Tables mashup using data from the project database. They also had to write a business proposal and report on their site's analytics. In other classes, students were tasked with writing articles, reading responses and essays in addition to the tasks related to the collaboration.

While subsequent classes followed a similar template of joint class meetings, presentations, a field trip and parallel activities, refinements were made based on student feedback and the observations of the instructors and the evaluator. One modification was to have the initial class meeting be between the students, with the guest speakers coming to the second meeting. In this initial meeting, the students explored the existing software system, tested its functionality, and began brainstorming ideas for improvements and questions for our subject-matter experts.

In addition, students are given worksheets for each meeting outlining that meeting's purpose and tasks. These worksheets must be completed and submitted that day. Rather than have the students work in teams in their respective classes, we now place students in teams that cross classes. Code and documentation are shared via a GitHub repository.

The journalism students also have created a folder on Google Drive containing information that is not yet incorporated into the system. This includes a database of expert sources, and a database containing information on the historical uses of various properties. The area where our college is located was once a thriving industrial town with factories producing pottery, rubber and iron, among other products. Students researched the environmental hazards associated with each type of manufacturing. They pored over old city directories to locate the sites of these businesses. In some instances, they had to search for the companies on Sanborn Fire maps to get precise address information. Then we generated Google street view images of the locations today. Once all of the data are cleaned, our intention is to mashup these locations with our original brownfield inventory. Where there are matches, we expect this will offer additional information about the hazards that might lurk below the surface of these sites. Where there isn't a match, we hope that our method will give stakeholders an idea of what might be there.

The projects that the students chose to work on during the Spring 2015 semester include substantial roles for both sets of students. One group worked on making the map module of the system more robust. The journalism and media students wrote explanatory content to help users interpret the pollution information. Another group enhanced the section of the site intended to help users find out about environment-related bills in the state legislature. Yet another group has devised a module that aggregates Twitter conversations about brownfield remediation and restoration, sorting them geographically.

One of the challenges of doing this work with undergraduates in an institution with no graduate students is that each semester, students are tasked with learning about the issues at the same time that they are learning the journalism or computer science content and skills. Despite this, we are moving closer to having a functioning information system.

Research methods and analysis

Initially, we intended to gauge changes in students' computational thinking using both self-assessments and anonymized data on student performance on assignments related to the computational thinking learning goals for each course. We were unable to obtain IRB approval for use of the student performance data. However, substantial scholarship supports the notion that learners with a high level of computing self-efficacy are more motivated to learn new computing methods (Moores, Chang, & Smith, 2006). Self-efficacy is defined as a belief that one can master a new set of tasks or skills and achieve desired ends. (Kvasny, Joshi, & Trauth, 2011).

Self-assessments were administered in pre-tests and post-tests over four semesters, to all students in five designated Computer Science courses and five designated journalism courses. All but one of the journalism courses was cross-listed with Interactive Multimedia. In Fall 2013, two courses from each discipline were included in the study; in the three other semesters, only one course from each discipline was included.

Pre-tests were administered to all students in person as paper surveys on the first day of class, following an explanation of the project and a review of human subjects' protection. Post-tests were administered electronically using Qualtrics software, with email invitations sent to students' official campus email accounts. Students were invited to complete the post-test during the last week of class, but were able to complete these into the "reading period" before final exams. Each semester, multiple follow-up reminders were sent to non-respondents, both from the independent evaluator and by the faculty member teaching each course.

Pre-test data was collected from 153 individuals. Post-test data was collected from 138 individuals, indicating a response rate of 90.20%. Attrition results both from students dropping the classes after the first day, or individual students declining to participate in the post-test. We did not collect data on race or gender demographics because of the small numbers of students from underrepresented backgrounds in the various classes.

All analysis was completed using SPSS software.

The total number of participants for which we have at least some pre-test and post-test data is 138: 49 from Science (35.51%), 36 from Arts & Communication (26.09%), 27 from Engineering (19.57%), and 26 from Humanities & Social Science (18.84%) (Figure 1).

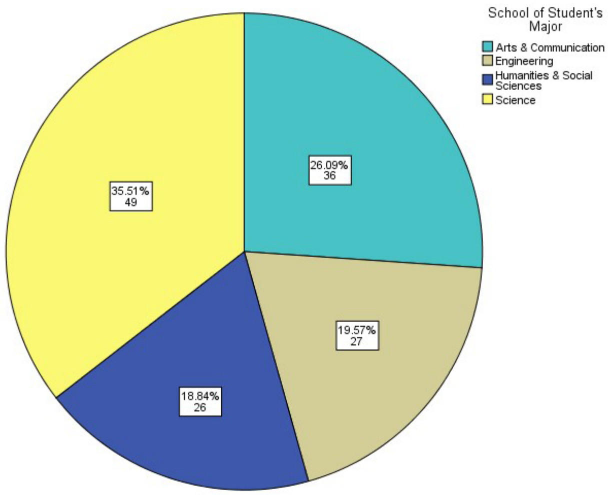

Figure 1. Distribution of participating students by school of student's major.

Computational Thinking Self-Assessment: Operationalization

All students were asked a series of eight questions, derived from ABET's General and Program-Specific (Computer Science) Student Outcomes Criteria for Accrediting Computing Programs (ABET, 2011, p. 3; items were derived from Criteria from 2012-2013 Accreditation Cycle Document, but these had not changed as of the most recent 2017-2018 document). The last three items indicated below are based on Dr. Jeannette Wing's definitions of computational thinking which have been widely distributed in a variety of publications (c.f. Wing, 2011). For journalism, items were added that were derived from the Accrediting Council on Education in Journalism and Mass Communications (ACEJMC) Accrediting Standards on Curriculum and Instruction (ACEJMC, 2012).

On both pre-test and post-test, students were asked "to what extent do you agree or disagree with each of the following:" with response categories of: Strongly Agree (coded 4), Agree (3), Disagree (2), or Strongly Disagree (1) with the following items:

- I can apply knowledge of computing appropriate to my major.
- I can analyze a problem, and then identify and define the computing requirements appropriate to its solution.
- I understand the impact of computing on society.
- I can use current computing techniques, skills, and tools necessary in careers for which my major prepares me.
- I can collaborate with others to design and develop computer based tools and technologies appropriate to careers for which my major prepares me.
- I can use abstractions.
- I can use logical thinking.
- I can use algorithms.

There are between 107-113 valid cases for each of these items.

Reliability analysis of these items as a measure of computational thinking is indicated by a Cronbach's alpha of .806 for pre-test items and .911 for post-test items. In other words, these items together are a reliable measure of the underlying concept of computational thinking.

Change from pre-test to post-test was computed as the mean arithmetical difference (i.e., mean of all pre-test minus post-test differences).

Computational Thinking - Journalism Students

Derived from ACEJMC standards, students in journalism classes were asked "to what extent do you agree or disagree with each of the following:" with response categories of: Strongly Agree (coded 4), Agree (3), Disagree (2), or Strongly Disagree (1) with the following items:

- I can conduct research and evaluate information by methods appropriate to journalism.
- I can edit.

Change from pre-test to post-test was computed as the mean arithmetical difference (i.e., mean of all pre-test minus post-test differences). It should be noted that an error in the electronic post-test made evaluation of the final item impossible.

There were 31 valid cases for the valid item.

Reliability analysis of the one valid item above added to previous eight items as a measure of computational thinking among JPW/IMM students is indicated by a Cronbach's alpha of .824 for pre-test items and .894 for post-test items.

In Fall 2014, an additional set of items was pre-tested by the researchers for journalism students, asking "to what extent do you agree or disagree with each of the following:" with response categories of: Strongly Agree (coded 4), Agree (3), Disagree (2), or Strongly Disagree (1) with the following items:

- I am motivated to learn new applications in computer technology on my own that are relevant to careers in journalism.
- I am motivated to learn new applications in computer technology with my peers that are relevant to careers in journalism.
- I am motivated to take courses in computer science that are relevant to careers in journalism.
- I am motivated to learn how computer technology is created for use in journalism.

Because of the small number of responses on these questions, these items are not analyzed in this study.

Overall, all students rated their own knowledge at the end of the semester higher in all items when compared to their self-ratings at the beginning of the semester, with a full increase of one unit or more in four of eight items (Figure 2).

Journalism students rated their own knowledge at the end of the semester higher in all items when compared to their self-ratings at the beginning of the semester, with nearly a full increase of

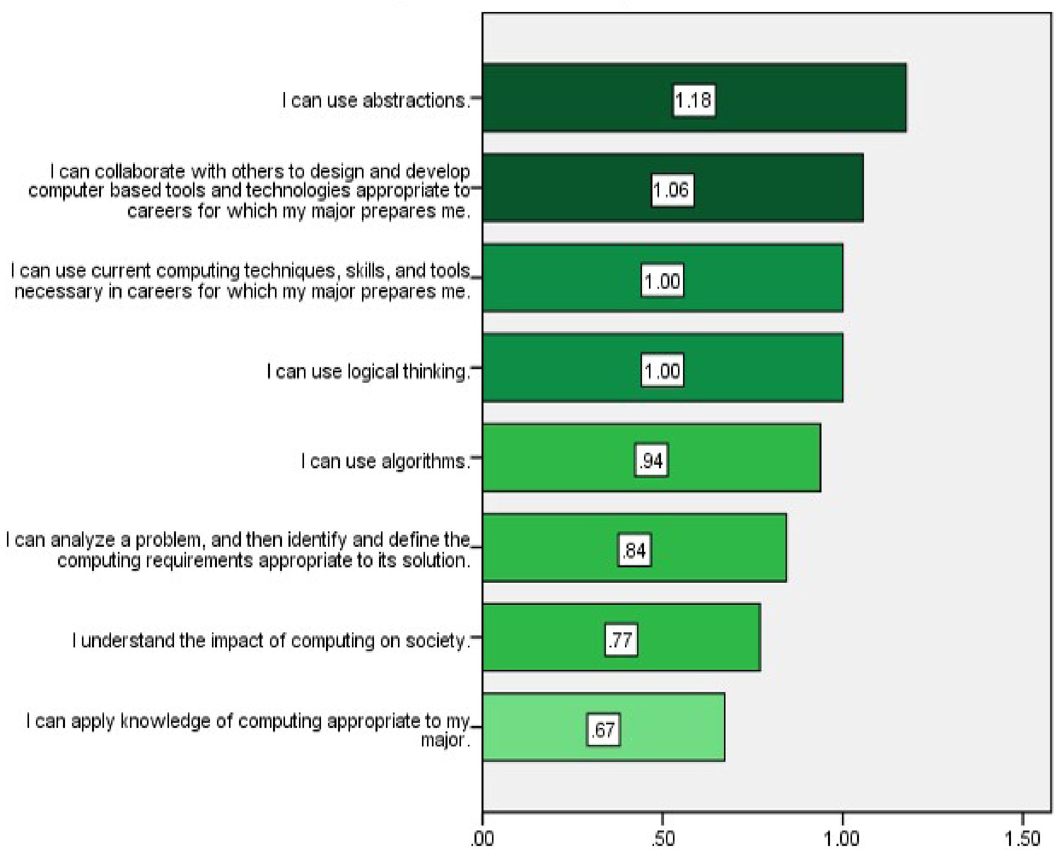

Figure 2. Mean change in all students, all semesters

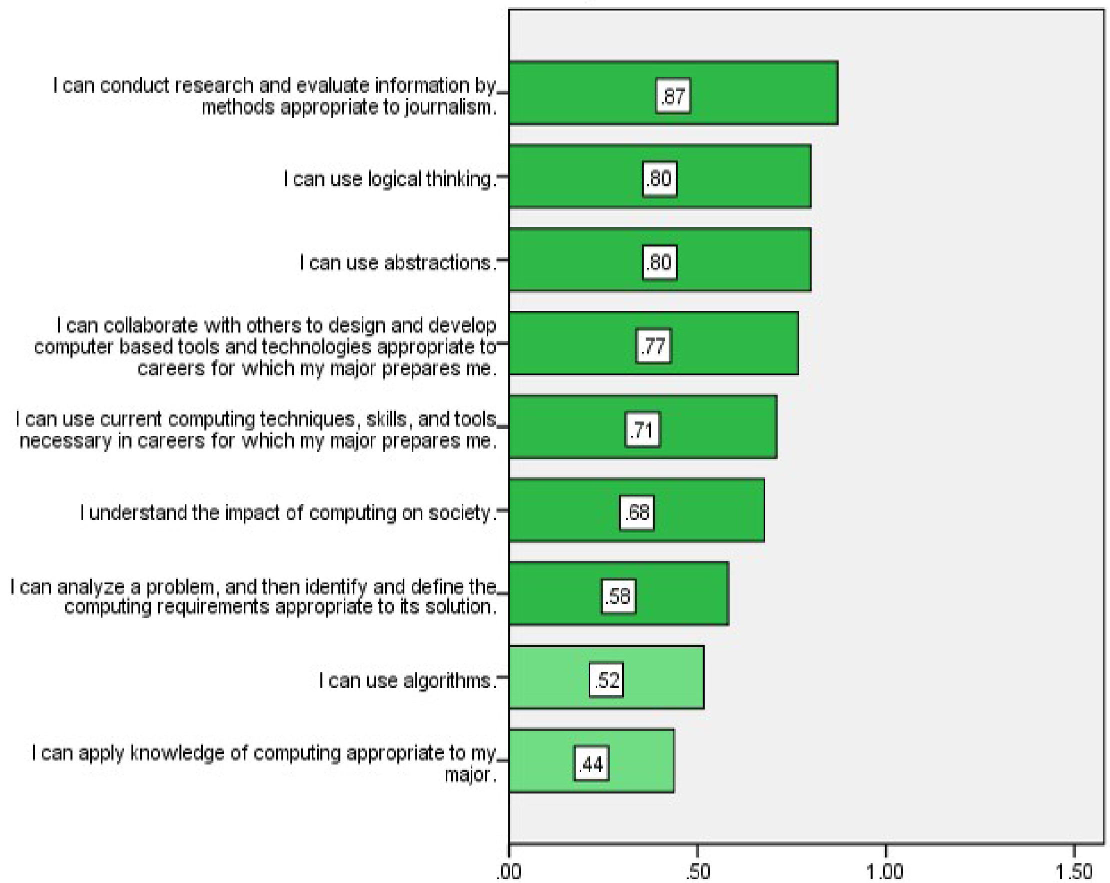

Figure 3. JPW/IMM students, all semesters

one unit in four of nine items and at least a half unit increase in all but one item (Figure 3).

Communications majors (*n* = 36: Digital Arts = 18, Interactive Multimedia = 16, Communication Studies = 2) indicated greater change in most of the items in comparison to their counterparts from the School of Humanities & Social Sciences (*n* = 26: Journalism & Professional Writing = 22, English = 3, Psychology = 1) (Figure 4). The one exception to this is in the item on research and evaluation, where HSS students indicated substantially greater change (NOTE: This is the ONLY item that is derived entirely from the ACEJMC standards). One-way analysis of variance determined that there are only two items with statistically significant difference: I can analyze a problem, and then identify and define the computing requirements appropriate to its solution ($F(1, 39) = 5.053$, $p = .030$) and I can use algorithms ($F(1,39) = 10.611$, $p = .002$). One other item is nearly significant: I can conduct research and evaluate information by methods appropriate to journalism ($F(1, 25) = 3.250$, $p = .083$). This graph suggests that A&C majors are consistently indicating greater gains in computational thinking than the HSS students (who are mainly journalism students) enrolled in the same classes; in some cases this is significantly more.

Conclusion

Journalism practice and study has been transformed by computer science. A broad consensus is forming that aspiring journalists need, minimally, to be competent data journalists who can use computing tools to extract, analyze and represent data in engaging ways. They also need to be able to collaborate with computational journalists in the creation of new journalism tools and artifacts. At minimum, this requires the ability

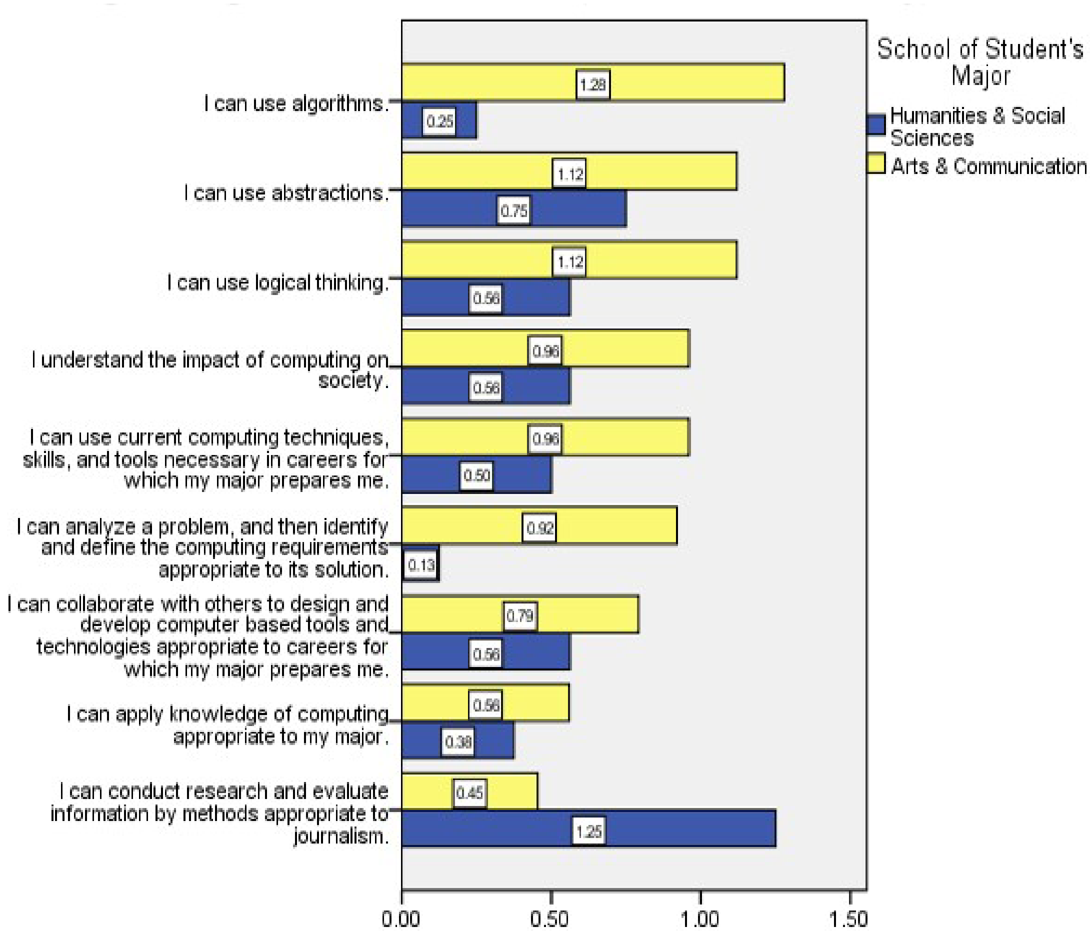

Figure 4. Change among A&C and HSS students, JPW/IMM classes only, all semesters

to analyze problems and identify the computing requirements for their solution.

Journalism educators are faced with the challenge of determining which aspects of computer science are most relevant to journalism study and practice, how they might be effectively integrated into an already-crowded undergraduate curriculum, and how to create a learning environment in which students feel confident in their ability to learn computational journalism.

The CABECT project was conceived in the belief that students will be more motivated to become proficient in the application of computing methods and tools to their fields of study if they are immersed in interdisciplinary collaborations with a community partner. Four semesters of data from a pilot collaboration among computer science, journalism and interactive multimedia students indicate that such collaborations can strengthen students' confidence in their understanding of the relevance of computer science to their fields.

Journalism students participating in the CABECT study report significant gains in their ability to analyze problems and define computing requirements relevant to solving them, and in their ability to collaborate with others in developing computing tools for journalism. While all students participating in the CABECT study report gains in computational thinking, journalism students lag behind their counterparts in Interactive Multimedia and Computer Science. This is a point worthy of further investigation.

One possible explanation for this discrepancy might be that the journalism students came to the classes with less prior computing instruction. As noted earlier, there are two required courses in the current journalism curriculum with explicit instruction in computing techniques: Writing for Interactive Multimedia and Computer-Assisted Reporting (renamed Data Journalism). As the data below shows, only two journalism students had taken CAR prior to taking the classes involved in the collaboration.

- Writing for Interactive Multimedia: 13/18 (IMM and other A&C majors), 10/26 (JPW and other HSS students)
- Writing for Interactive Multimedia: 13/18 A&C, 10/26 HSS
- Computer-Assisted Reporting: 0/18 A&C, 2/26 HSS

We did not collect information on the prior class enrollments of the interactive multimedia students. Because it is cross-listed with journalism, we know that most of these students have at least taken the Writing for Interactive Multimedia class. It is reasonable to assume that a significant portion of IMM students also have taken at least the required introductory programming courses in that major - Introduction to Interactive Computing and Design Fundamentals for the Web. The courses also explicitly introduce computing concepts such as automation, algorithms and abstractions.

Anecdotally, the IMM students have been quick to volunteer to help with html and php scripting for their team's software modules. A student who had taken several journalism and IMM classes took the lead in helping the computer science students design a blog module; she had more experience with blogs and content management systems than they. On the other hand, some journalism students have expressed an interest in becoming more adept at web scraping, data visualizations and other computing techniques for journalism.

Limitations

While our results show promise, several limitations limit its broad applicability. First, our sample sizes are not as large or diverse as we would like. Second, IRB restrictions kept us from pursuing our original intention to compare student post-test responses to their performance on assignments designed to meet the course learning goals related to computational thinking. Finally, we look forward to seeing our model replicated and tested by other faculty at other colleges.

Conclusions

We conclude, therefore, that the CABECT model shows promise as a context for deepening journalism students' engagement with computational thinking. In the next year, we expect the CABECT model to be adopted by other faculty members in journalism and other disciplines, so we will be able to see whether our results will be replicated. Comparisons between responses from journalism and interactive multimedia students suggest that the explicit programming instruction in the Interactive Multimedia curriculum might help explain those students' greater gains. (It's also possible that the IMM students were already more comfortable with computing, giving the technological focus of that major.)

Directions for future research include exploring the degree to which instruction in web design, which is mandatory for our journalism majors, might be enhanced to incorporate more explicit concepts related to computational thinking. This, in turn, might allow journalism students to participate in interdisciplinary computing collaborations with greater confidence, and also fuel their desire for further study. Research by Park and Wiedenbeck (2011) on computational thinking in introductory web development courses suggests this approach may hold promise. They observe:

> [W]e have identified a set of computational concepts that permeate the challenges students encounter when learning web development. Notation puts students into the mindset of instructing computers using carefully specified language. Hierarchies and paths offer ways of thinking about familiar systems such as file systems and the web, while setting

> the stage for more advanced topics like traversing the JavaScript Document Object Model (DOM). Nesting makes frequent appearances in HTML, giving students practice with navigating multiple levels of nested code. Through HTML, students can become familiar with the notion of parameters and arguments. By separating content (HTML) from presentation (CSS) and behavior (JavaScript), students apply decomposition and abstraction in order to manage complexity. Through an elementary web development course, these computational concepts are introduced in simple, concrete forms and lay the groundwork for further learning. (p. 131)

In addition, faculty at several institutions with more diverse student populations are considering adopting the CABECT model for their own computational journalism collaborations. This will help us understand whether this is an effective strategy for helping student journalists of color, particularly, become more computationally fluent.

The CABECT model demonstrates that collaborative learning holds promise for motivating journalism students to become more deeply engaged with computing as it pertains to their major. If journalism students are able to enter these collaborations with at least introductory-level proficiency in web design and a grasp of the computing concepts embedded in that activity, they might be able to participate in such collaborations with greater confidence and to greater effect. If further research supports the efficacy of the CABECT model, journalism and computing educators will have a potent tool for addressing the critical shortage of computing professionals in and beyond the news industry.

References

1. ABET. (2011). *Criteria for accrediting computing programs: Effective for reviews during the 2012-2013 accreditation cycle*. Baltimore, MD: ABET. Retrived from http://www.abet.org/wp-content/uploads/2015/04/cac-criteria-2012-2013.pdf
2. Accrediting Council on Education in Journalism and Mass Communications (ACEJMC) (2012). *ACEJMC Accrediting Standards*. Retrieved from https://www2.ku.edu/~acejmc/PROGRAM/STANDARDS.SHTML
3. Bradshaw, P. (2012a, August 2). Two reasons every journalist should know about scraping. *Journalist.co.uk*. Retrieved from https://www.journalism.co.uk/news-commentary/-two-reasons-why-every-journalist-should-know-about-scraping-/s6/a550001/
4. Bradshaw, P. (2012b, August 13). How to teach a journalist programming. *Data Driven Journalism*. Retrieved from http://datadrivenjournalism.net/news_and_analysis/how_to_teach_a_journalist_programming
5. Coddington, M. (2014). Clarifying journalism's quantitative turn: A typology for evaluating data journalism, computational journalism, and computer-assisted reporting. *Digital Journalism, 3,* 331-348. doi:10.1080/21670811.2014.976400
6. Ericson, B. (2014a). *AP data for the United States, 1998-2014* [Data file]. Retrieved from AP Data for the United States, 1998-2014: https://home.cc.gatech.edu/ice-gt/588
7. Ericson, B. (2014b). *Detail race and gender information 2014* [Data file]. Retrieved from AP Data for the United States, 1998-2014: http://home.cc.gatech.edu/ice-gt/590
8. Flew, T., Spurgeon, C., Daniel, A., & Swift, A. (2012). The promise of computational journalism. *Journalism Practice, 6*(2), 157-171.
9. Guzdial, M. (2014, March 4). AP CS 2014 results: Big jumps in participation! Demographics still poor [Computing Education Blog]. Retrieved from https://computinged.wordpress.com/2015/03/04/ap-cs-2014-results-big-jumps-in-participation/

10. Gynnild, A. (2014). Journalism innovation leads to innovation journalism: The impact of computational exploration on changing mindsets. *Journalism, 15*(6), 712-730.
11. Hernandez, R. (2011, August 13). Why all your students must be programmers: the #AEJMCBattleRoyale [Web Journalist Blog]. Retrieved from Webjournalist: http://blog.webjournalist.org/2013/08/11/why-all-your-students-must-be-programmers-the-aejmcbattleroyale/
12. Hewett, J. (2014, November 7). Data journalism, computer-assisted reporting and computational journalism: what's the difference? Hackademic: Journalism Education (and More). Retrieved from http://hackademic.net/2014/11/12/data-journalism-computer-assisted-reporting-and-computational-journalism-whats-the-difference/
13. Johnson, M. (2012, March 19). On building a beter journalist: A guest post by Michelle Johnson. *Amy Alexander Community Forum*. Retrieved from http://amyalexanderink.blogspot.com/2012/03/what.html
14. Kvasny, L., Joshi, K. D., & Trauth, E. (2011). The influence of self-efficacy, gender stereotypes and the importance of it skills on college students' intentions to pursue IT careers. *Proceedings of the 2011 iConference (iConference '11)*, 508-513. Retrieved from http://doi.acm.org/10.1145/1940761.1940831
15. Margolis, J. (2008). *Stuck in the shallow end: race, education and computing.* Cambridge MA: MIT Press.
16. McAdams, M. (2012). Teaching programming to journalists. *Teaching Online Journalism*. Retrieved from http://mindymcadams.com/tojou/2012/teaching-programming-to-journalists/
17. Moores, T.T., Chang, C.J., & Smith, D.K. (2006). Clarifying the role of self-efficacy and metacognition as predictors of performance: construct development and test. *ACM SIGMIS Database: the DATABASE for Advances in Information Systems*, *37*(2-3), 125-132. Retrieved from http://doi.acm.org/10.1145/1161345.1161360
18. National Research Council of the National Academies, Committee for the Workshops on Computational Thinking, Computer Science and Telecommunications Board, Division on Engineering and Physical sciences. (2010). *Report on a workshop on the ccope and nature of computational thinking.* Washington, DC: National Academies Press.
19. Park, T. H., & Wiedenbeck, S. (2011). Learning web development: Challenges at an earlier stage of computing education. ICER '11 Proceedings of the Seventh International Workshop on Computing Education Research, pp.125-132. Retrieved from http://dl.acm.org/citation.cfm?doid=2016911.2016937
20. Pulimood, S. M., & Wolz, U. (2008). Problem-solving in community: A necessary shift in CS pedagogy. *SIGCSE '08: Proceedings of the 39th Technical Symposium on Computer Science Education,* 210-214.Retrieved from . http://dl.acm.org/citation.cfm?id=1352209
21. Pulimood, S. M, Shaw, D., & Lounsberry, E. (2011). Gumshoe: A model for undergraduate computational journalism education. *SIGCSE '11 Proceedings of the 42nd ACM technical symposium on Computer science education,* 529-534. Retrieved from http://dl2.acm.org/citation.cfm?id=1953314
22. Royal, C. (2012, October 9). Journalism schools need to get better at teaching tech where the girls are. *NiemanLab*. Retrieved from http://www.niemanlab.org/2012/10/cindy-royal-journalism-schools-need-to-get-better-at-teaching-tech-where-the-girls-are/
23. Royal, C. (2013, August 26). We need a digital-first curriculum to teach modern journalism. *Media Shift.* Retrieved from http://www.pbs.org/mediashift/2013/08/we-need-a-digital-first-curriculum-to-teach-modern-journalism/
24. Royal, C. (2014, April 28). Are journalism schools teaching their students the right skills? *NiemanLab*. Retrieved from http://

www.niemanlab.org/2014/04/cindy-royal-are-journalism-schools-teaching-their-students-the-right-skills/

25. Royal, C. (2015, March 25). Why universities need to embrace coding across the curriculum. *Media Shift*. Retrieved from http://www.pbs.org/mediashift/2015/3/why-universities-need-to-embrace-coding-across-the-curriculum
26. Stray, J. (2013). Introduction: computer science and journalism, JMSC6041: Computational Journalism [Syllabus]. Pokfulam Road, Hong Kong: Journalism and Media Studies Centre, The University of Hong Kong. Retrieved from http://web.archive.org/web/20131026113731/http://courses.jmsc.hku.hk/jmsc6041spring2013/2013/02/14/introduction-computer-science-and-journalism/
27. Wihbey, J. (2015, March 11). Addressing journalistic innumeracy and math phobia: A sequence of resources to sharpen your mind. *Journalist's Resource*. Retrieved from http://journalistsresource.org/studies/society/news-media/journalistic-innumeracy-math-phobia-sequence-resources-to-sharpen-your-mind#
28. Wing, J. (2011, March 6). Computational thinking: What and why? *The Link: The Magazine of the School of Computer Science at Carnegie Mellon University*. Retrieved from http://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why