Getting CS Undergraduates to Communicate Effectively

Andreas Karatsolis, Iliano Cervesato, Nael Abu-Ghazaleh, Yonina Cooper, Khaled Harras, Kemal Oflazer, Thierry Sans
Carnegie Mellon University – Qatar campus
akaratsl@qatar.cmu.edu, iliano@cmu.edu, nabughaz@qatar.cmu.edu, yonina@cs.cmu.edu, kharras@cs.cmu.edu, ko@cs.cmu.edu, tsans@qatar.cmu.edu

Background

Computer Science programs have recently started to introduce a sophomore or junior level course on Professional Communication, in hope that students will understand the basic principles of communication and apply them in later courses or their careers. However, these courses are often driven by lower-level concerns, mostly pertaining to correctness, concision, coherence or document design, falling short of giving students the skills to carry out communication “performances” in a professional setting. However, all professional organization statements and research in learning express the need of engaging students with real audiences in real contexts through professional genres

> “mastery of the discipline includes not only an understandingof basic subject matter, but also an understanding of the applicability of the concepts to real-world problems”
> (ACM IEEE/CS Computing Curriculum 2008)

Developing the Model

Our immediate next goal is to formalize this grassroots effort into a Communication across the Curriculum (CAC) model, following the tripartite model of mastery. This process requires

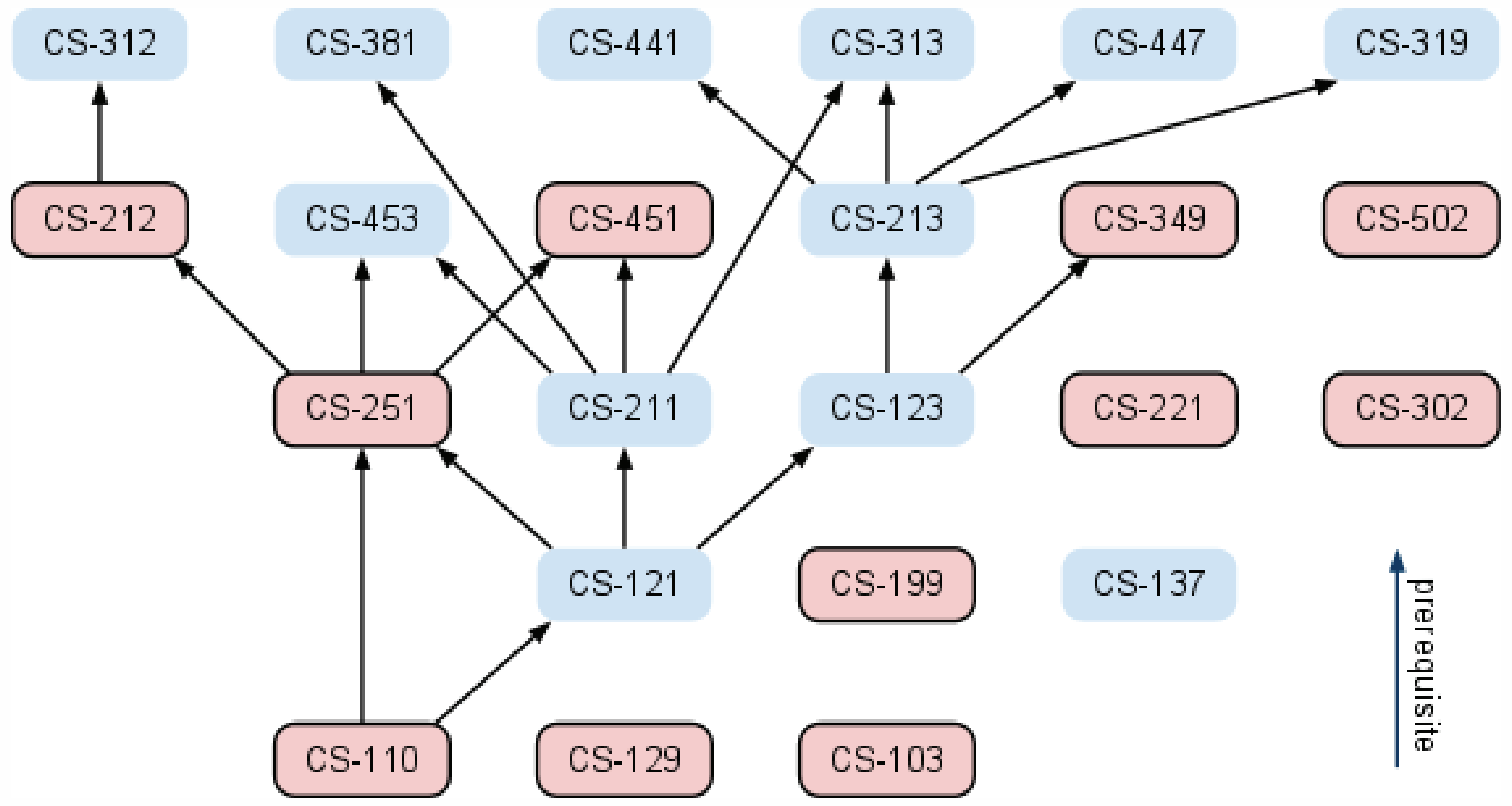
- A curriculum sequencing exercise aligning the learning objectives of courses with communication skills components.
- A series of workshops for the faculty to introduce the vocabulary of professional communication and rhetoric.
- Development and alignment of evaluation methods, ideally the same or similar assessment rubrics.
- A longitudinal assessment effort to evaluate the effectiveness of this model in supporting professional communication practices through employer and alumni surveys
- An iterative review plan for the CAC model to make it sustainable throughout the four years of the CS program.

The benefits of developing such a model are:

- Instruction and practice in communication skills will not be limited to one course, but can be sustained throughout the curriculum
- Students will see from the beginning the relevance of all the class room activities and assignments to professional CS practices
- Faculty will be able to engage students in real projects early and often

Abstract

Professional organizations, such as the ACM and the IEEE, have recognized that the communication skills students develop in the classroom do not always match workplace expectations. For many programs, the simplest option was to add a dedicated technical communication course to their curriculum, while others attempted to embed writing in several courses. This poster presents our efforts over the past two years to explore a communication-across-the-CS-curriculum model where many “traditional CS” courses have added communication components, in addition to revamping our technical communication course to expose students to authentic disciplinary practices. The results, so far anecdotal, point to improved student performance and attitudes across several communication dimensions, in particular writing and presentation.



Legend for Figure 2

CS-103: Fundamentals of Computation
CS-110: Introduction to Programming
CS-129: Freshman Immigration
CS-121: Introduction to Data Structures
CS-123: System Skills in C
CS-137: Introduction to Web 2.0
CS-199: Discovering Logic
CS-211: Fundamental Data Structures and Algorithms
CS-212: Principles of Programming
CS-213: Introduction to Computer Systems
CS-221: Technical Writing
CS-251: Great Theoretical Ideas in Computer Science
CS-302: Technology Field Research in Developing Communities
CS-312: Foundation of Programming Languages
CS-313: Foundations of Software Engineering
CS-319: Introduction to Cloud Computing
CS-349: Introduction to Computer and Network Security
CS-381: Artificial Intelligence
CS-441: Computer Networks
CS-447: Computer Architecture
CS-451: Algorithm Design and Analysis
CS-453: Formal Languages and Automata
CS-502: Technology and Global Development

Mastery

Practicing Technique

The development of expertise in most fields requires a certain degree of technical skill which can only come through sustained practice. In music or in sports, it takes months or years of practicing technical skills before one can perform in a concert or a game. Similarly, we decided that practicing technical writing and speaking skills was a critical step in enacting this framework. Most technical communication courses make this the focal point of the work that students do, but in our case we decided that recitation sessions or online modules would allow students practice without taking up classroom time. These practice sessions were directly related to the patterns they were encountering in classroom discussions of different genres.

Engaging in Simulation

The second step towards developing expertise with communication skills is to begin simulating the practice. Students work on projects where they have to simulate practice: for example, they have to develop job application materials in response to a job or internship advertisement, or they have to write a proposal on a technical project in response to a request for proposals the instructors create for a real organization. Students, therefore, have to understand the affordances and constraints of real genres and produce responses for real audiences, with the exception that these audiences do not receive the materials or act upon them.

Performing in Real Contexts

The last step of the model requires students to engage with real audiences (“clients”) and perform in communication events which are evaluated both by the instructors for the purposes of the course, but also by the clients. Students are introduced to real communicative performance through a user guide assignment. As user manuals are rarely targeting specific audiences, it is critical for some clients to find people who can help them rewrite or develop from scratch user guides for their constituents. For example, in the Fall 2010 semester students developed user guides for tools and technologies intended to assist people with disabilities. Their client is MADA, the newly formed center for assistive technologies in Qatar, but their audience (who they will have to interact with for feedback and requirements specifications) is disabled people. This interaction makes students accountable for the quality of documentation they are producing. At the same time, the clients (MADA in this case), expect artifacts they could utilize with little or no revision.

Figure 1: Mastery model, adapted from Ambrose et al (2010), How Learning Works: Seven research-based Principles for Smart Teaching