Integrating an Intensive Experience with Communication Skills Development into a Computer Science Course

Lori Pollock
Computer and Information Sciences
University of Delaware
Newark, DE 19716
pollock@cis.udel.edu

Abstract

This paper describes how a technical computer science course was transformed into an intensive communication skills course without sacrificing the technical content of the course. By integrating this experience into existing technical courses, the acquired skills are specific to the CS context without requiring an additional course. The main contribution of this paper is a set of activities which are targeted to building communications skills required for successful research in computer science at any level, but also generally useful for computer science students entering careers not involving basic research. We describe the specific methods and tools implemented in a way to provide considerable support, guidance, and feedback to students without a large investment by the professor.

1 Introduction

While the main role of an advisor of a senior thesis, masters or Ph.D. student is to teach the necessary skills for performing high quality research and to mentor the student in their first research experience, advisors often spend considerable time teaching basic communication skills. Computer science students at all these levels typically lack experience and skills in reading and critiquing technical papers, performing thorough literature searches, organizing and writing a solid research report or proposal, organizing and presenting an oral or poster presentation, designing an experiment to evaluate their theoretical contributions, and openly defending their own ideas and opinions. Unfortunately, an English or communications course offered by another department is not adequate as these skills are best learned in the context of computer science in order to address the specific issues within our discipline. One way to address this deficiency is to offer a separate course that focuses on developing and evaluating student communication skills in a discipline-specific way for computer science[2]. However, many computer science curricula do not have room or the resources for an additional non-technical course of this kind.

This paper describes how a technical computer science course was transformed into an intensive communication skills course without sacrificing the technical content of the course. The main contribution of the paper is a set of activities which are targeted to each of the communications skills mentioned above, and a description of how they were implemented in a way to provide considerable support, guidance, and feedback to students without a huge investment by the professor. While this course is a graduate level course, we believe that this same kind of transformation could be applied directly or be easily adapted to many different upper level undergraduate computer science courses.

2 Pitfalls of a Traditional CS Program

The majority of CS lecture courses are taught using a passive learning style in the classroom and hands-on programming and homework assignments outside the classroom. One positive effect of classes that incorporate active and group learning techniques[1, 5, 3] is the nurturing of the student's oral communication skills. Practice in written communication skills is promoted in courses that include term projects and papers. However, the primary source for fostering communication skills in a CS program is usually the seminar courses. These courses have primarily small class sizes, the students are challenged to read technical papers as opposed to a textbook, and the professor and students take turns leading discussions. In addition, these courses usually include more open-ended research projects and papers.

Our past experience with this organization of a seminar has revealed several deficiencies. Students who are not presenting on a particular day are often ill prepared to participate in the discussions. This ill preparedness results in monotonous class periods and limited discussions. Poor student presentations greatly affect the other students' learning. Student term papers with a single final deadline fall short of the learning objectives of these assignments. These observations led us to examine how seminar courses could be redesigned to achieve more extensive learning objectives related to building communication skills.

3 Transformed Course Learning Objectives

The particular course which we targeted for the first transformation is a seminar course on Advanced Program Analysis and Transformation. In this course, we explore static program analysis and machine-independent code transformations performed in modern optimizing compilers. Topics include control flow analysis, intraprocedural and interprocedural data flow analysis, alias analysis, advanced program representations, and classic code-improving transformations. The course textbook is a set of technical papers from conferences and journals. The prerequisite is a basic compiler construction course. The course meets twice a week for 1 1/2 hours each class meeting.

With the revised learning objectives to emphasize communication skills, students completing the transformed course should significantly improve their ability to independently and successfully:

- Perform a thorough bibliographical search of a specific computer science topic.
- Categorize papers on a particular topic and identify the most important papers to focus their reading.
- Write a thorough, critical review of a technical paper.
- Compare and summarize a set of technical papers in a specific area, including construction of a historical perspective on the research contributions, and extraction of the important problems, contributions and current limitations of the state-of-the-art.
- Write an informative, but concise and critical, survey of the state-of-the-art in an area of computer science.
- Identify open research problems.
- Participate actively in a group brainstorming session focusing on developing new approaches to a problem.
- Develop an experimental plan which would evaluate the effectiveness of new theoretical ideas.
- Write a well developed, concise research proposal.
- Defend one's own research ideas and the state-of-the art on a topic, through organization and presentation both orally and visually.

4 Course Activities

Communication skills were stressed in both the classroom activities and the projects on which the students worked outside the classroom in groups.

4.1 Classtime Organization

The goals of the transformed classroom sessions were to summarize the important aspects of the technical papers in a way that involved active presentation by at least 4-5 students during each class period, high quality presentations, and active participation in discussion of the issues and solutions of each paper by the whole class. Our approach to achieving these goals consisted of (1) student minilectures, (2) a revised faculty role, (3) immediate speaker feedback, (4) special review sessions, and (5) a set of tools for class preparation.

Minilectures, Teams, Faculty Role. Each classroom session concentrated on one or two related technical papers. Each paper was partitioned for presentation by 4-5 class members, thus a team was created for each paper. A common partitioning for a paper was:

1. introduction of the problem and motivation
2. overview of the basic approach of the paper
3. part of the details of the approach
4. more details of the approach
5. restrictions, extensions, applications of the work
6. putting the work into perspective with related work

Approximately 2 weeks ahead of a given session, the students were asked to volunteer to be responsible for one of these portions of the paper. While students were able to volunteer on a first come, first serve basis, each student knew they would have to volunteer for at least one presentation every other class period, about once a week. A tally of the number of presentations by each student was kept, and a difference of no more than one presentation among any two students was maintained throughout the semester.

Each presentation was planned to be 10-15 minutes depending on the content, and presented with handwritten or typed overhead slides. In order to give each class member a complete set of notes for the course, each presentation was xeroxed for the whole class, typically before the class presentations, but sometimes afterwards.

Each speaker was encouraged to meet with the instructor when they had difficulty understanding the concepts they were scheduled to present. Speakers were given audience and instructor feedback for the first few presentations by having each class member complete an evaluation form, which was summarized anonymously and returned to the presenter. Later, a presenter could request evaluations and feedback from the instructor if desired.

While the professor also took turns with the minilectures, the faculty role during classtime consisted primarily of interjections to clarify when important points were not emphasized adequately or students had questions. The presenter was not given the added responsibility of being the sole person to answer questions on the material covered by their presentation. Students were encouraged to try to answer the other students' questions.

This scheme for presentations of technical papers had a number of positive effects. In a single class period, there were at least 5 minilectures by different people. This broke the monotony of a single presenter, and created natural breaking points that changed the flow. Students came to class much more knowledgeable of and prepared to discuss the papers, because even if they were only responsible for one part of the paper, they were uncomfortable only reading and presenting their piece, without knowing how it related to the rest of the paper. Students gained significant practice in front of the classroom. Each student prepared and delivered approximately 8 minilectures that semester. As the semester came toward the end, the students began to complement each other on how they were improving on their oral communication skills, and remarked on how much more comfortable they themselves were becoming with their own presentations. It appears that one gains more practice and self confidence from giving more presentations even when the presentations are significantly shorter. This scheme also distributed the workload for a given student over the entire semester, rather than being responsible for presenting and leading discussion for one or two complete class periods.

Special Review Sessions. In a seminar course, it is common to cover a large set of different papers and research contributions in a particular area. We found it extremely useful to set aside a full class period at least once a month to summarize and review the set of papers presented over the past month. However, this kind of session is most successful when active discussion results. Thus, we supported these sessions by developing worksheets to be completed by each student either individually or in groups, prior to the special review discussion session. Because these worksheets were designed to create discussion and bring the papers into perspective with one another, the review session became a very exciting time of reflection.

Technical Paper Review Form

Paper Citation:
Research/Project Goal(s):
Technical Details:
  Any unfamiliar terminology:
  Summary Overview:
  Discussion/Critique:
  Uses/application of this work:
Relevant References:
  Most relevant previous work on which it builds:
  Work by others which focuses on
  similar/same problem, but different approach:
  Work that uses or builds on these contributions:
Open Issues/Research Opportunities:
(restrictions, what was not done, what failed to do,
  where it doesn't work well)
Other Notes:

Table 1: Technical Paper Review Form

Tools for Class Preparation. Several tools were developed with the goal of helping students focus their attention on the important aspects of technical papers without getting lost in the minor details as they read, helping students to prepare for classroom discussions in an organized way, and increasing student participation in classroom discussion by increasing their comfort levels and self confidence.

The first tool was the technical paper review form. Students used this form as they read and reread technical papers. They would come to class with these completed forms for the papers to be presented in that class period, and also used these forms throughout their research projects. At first, students found these most useful in helping them organize their reading and gather their thoughts concerning a particular paper. As writing the summary sections became almost mechanical to them, the students devoted more time to the critique sections of the reviews. Thus, the forms began to work toward giving the students significant practice in writing critical reviews of others' research and writing. The review form is shown in table 1. Many of the students are continuing to use these paper review forms in other classes and in their own research. They can be exploited as a quick reference for the key points of a paper read awhile ago.

The second tool was the worksheet for summarizing and reflecting on a set of related research papers, with the goal of grasping a good intuitive picture of the state-of-

the-art in that area. These worksheets were used for the research projects to aid in making the step from individual paper reviews to a written literature survey. The papers summary worksheet consisted of the following:

Papers Summary Worksheet

1. What overall goals did the papers have in common?
2. List each paper and the specific goals it had within the overall goal.
3. Group the papers according to similar goals, and describe each approach, contribution, and restrictions to achieving that goal.
4. Brief summary of the state-of-the-art in chronological order.
5. Brief summary of open issues/research opportunities.
6. Other Notes.

These worksheets were also used in conjunction with the third tool, specialized worksheets, to prepare for the periodic special review discussion classes. The specialized worksheets were a set of 5-10 short answer questions that focused on basic understanding of key techniques covered since the previous review session, comparisons between methods, and advantages and disadvantages of different methods. The students found that having to complete the summary forms and worksheets helped them reflect on the papers and their relationships better than just knowing they were going to discuss them in class. They admitted to doing a much more thorough reexamination of the set of papers.

The last tool that was used was targeted toward getting the students to feel comfortable speaking in class and voicing their opinion. While the technical paper review forms encouraged students to think critically about the papers they were reading, rather than believing that any paper that was published must have no flaws or restrictions, students would often not speak up in class when it came to critiquing. Team building activities used in other settings were employed to break these barriers. An example of a team building activity that worked well was a day trip to a corn maze, where the class was challenged to work together to find their way out of the maze.

4.2 Experiencing The Research Process in Depth

While the classroom activities focused mostly on oral communication skills and reading and critiquing technical papers, the outside project focused on reading, critiquing, summarizing, and technical writing skills. The goal of the project was the experience of the process, not the final deliverable of a research proposal on some topic of interest. The final research proposal needed to include background on the state-of-the-art with a clear statement of the problem, proper citations, a detailed account of the research issues to be addressed, and a preliminary proposed approach with an evaluation plan for assessing the success of the approach.

Due	Phase: Deliverable	Effort	%
3-Sep	Choose topic/group	Group	0
9-Sep	1:Bib file: first round	Indiv	4
15-Sep	1:Bib file: 2nd round	Group	4
22-Sep	1:Annotated Bib Papers categorized	Group	5
22-Sep	2:Focus set identified	Group	4
12-Oct	2:Paper reviews: 1st half	Indiv	5
19-Oct	2:Paper reviews: 2nd half	Indiv	5
26-Oct	2:Revised paper reviews Summary of papers	Group Group	5
9-Nov	3:Lit Survey draft	Indiv	5
17-Nov	3:Lit Survey final	Indiv	8
22-Nov	4:Brainstorming report	Group	5
30-Nov	5:Proposal draft	Indiv	6
9-Dec	5:Proposal final	Indiv	14
7-Dec	5:Proposal oral	Indiv	10

Table 2: Research Process Schedule

The process from topic selection through written research proposal was partitioned into 14 deliverables and minideadlines, performed in five phases. Table 2 shows the phases, deliverables, and original deadlines. The phases are indicated by the following numbers on the tasks: (1) Bibliography Construction, (2) Paper Reviews, (3) Written Literature Survey, (4) Brainstorming, (5) Research Proposal Presentation. The schedule for the project had built-in time for giving feedback on various subtasks, and revised versions of different deliverables.

The project was also divided throughout the semester in terms of individual and group efforts. The tasks were carefully designated according to which method would lend itself to increased learning. Column 3 of the table indicates whether a task was performed as a project group or individually. The last column indicates the number of points assigned to each deliverable out of a total of 80 points designated for this entire project. A similar approach to a semester-long process focusing on communication skills is described by M. Michael at King's College[4]; our activities differ as ours include research-oriented skills, while theirs were incorporated into a semester-long programming project.

For all students, this was the first time they had been assigned a research project with the minideadlines and feedback, beyond a review of a first draft of a term paper. The feedback on a bibliographical search, and an opportunity to go back and revise and intensify their

search was a huge help to students. In addition, the students later drafts of both the literature survey and the research proposal were miles ahead of their first drafts in most cases. The large number of minideadlines caused students to spend much more time on the individual phases than they admitted they would have spent if there were only one or two deadlines near the end of the semester.

4.3 Investigating Infrastructures.

In addition to the research project, students also worked in groups on a compiler infrastructure investigation. The goal of this project was to gain the experience of the process of familiarizing oneself with a large software system in enough detail to be able to effectively determine whether the system would be appropriate to extend for one's own purposes. In compiler research, like many other fields, researchers need to build prototypes of their techniques to evaluate their effectiveness and efficiency. In order to focus the implementation effort on the new technique and avoid extraneous coding, the first step is to identify a compiler infrastructure that would ease the implementation, but be robust enough to perform solid experimental studies. This phenomenon occurs in many areas of computer science which involve empirical study through implementation.

For this project, the students selected one of the freely available compiler infrastructures, downloaded, installed, and played with running the system, read the documentation and some of the code, answered a set of questions directed toward judging the strengths and weaknesses of the system, and then prepared and presented a poster. We held a poster session one day in class in which the posters were displayed much like at a conference, and the poster presenters took turns explaining their poster to the class members as they walked around the room. The content of the posters was a visual description of the infrastructure to potential users. We then voted on the best poster presentation based on a set of criteria.

This was the first time many of the students had been given the task of examining a very large software system. Most students had only written their own programs that did not build on anyone else's programs, and had never examined or written a large system that included many directories, files, and lines of code. In this case, they were given specific questions for which the answers were buried in documentation or actual code. They found the exercise to be much like a puzzle, but in fact, it involved group discussions, reading and writing, and then preparation of a visual display.

5 Lessons Learned

The most noticeable effect of this course transformation was the significant improvement of student oral presentation skills over the semester. After leading seminars in the old style for many years, this was the first time that the improvement was so obvious for all students. The frequent opportunities to present in front of the class had really had a positive effect.

Another significant effect was how well the students were prepared for classroom discussion using the tools for classroom preparation and the partitioning of papers to make more students directly responsible for each paper. The increased classroom preparation led to much deeper discussions of the issues and proposed solutions, rather than high level discussions.

Many students commented on how the feedback and partitioning of group and individual work throughout the research project was invaluable. The resulting research papers were clearly the most thorough and involved more creative thoughts on proposed solutions than previous instantiations of this course. One aspect of the research project that needs some adjustment is the minideadlines. Some of the minideadlines were too close together, while others could have been shortened in order to lengthen the time for others.

Instructors are always concerned about how much time it will take to perform such a transformation to a course. The classtime organization involved taking 5 minutes at the start of each week to get volunteers for upcoming presentations. The most time was spent giving feedback on drafts of the literature surveys and the research proposals. However, the final products and lessons learned by the students were worth the time.

References

[1] Johnson, D. W., Johnson, R. T., and Smith, K. A. Cooperative Learning: Increasing College Faculty Instructional Productivity. *ASHE ERIC Higher Education Report No. 4* (1991).

[2] Kay, D. G. Computer Scientists Can Teach Writing: An Upper Division Course for Computer Science Majors. *Proceedings of ACM SIGCSE* (1998).

[3] McConnell, J. J. Active Learning and Its Use in Computer Science. *SIGCSE/SIGCUE Conference on Integrating Technology into Computer Science Education* (1996).

[4] Michael, M. Fostering and Assessing Communication Skills in the Computer Science Context. *Proceedings of the ACM SIGCSE* (2000).

[5] Silberman, M. L. *Active Learning: 101 Strategies to Teach Any Subject.* Allyn and Bacon, 1996.