A Course-based Undergraduate Research Experience (CURE) in Computer Science: An Experience Report*

Fahmida Hamid
Computer Science
Grinnell College
Grinnell, IA 50112
fahmida.hamid@gmail.com

Abstract

This article demonstrates a pedagogy of a research-focused Computer Science course for undergraduates in a liberal arts environment. The long term benefits of Course-based Undergraduate Research Experiences (CUREs) in different STEM fields are the driving forces for modeling such a course. The course balances a research-focused, semi-supervised, active learning experience and a traditional lecture-focused, supervised classroom-environment. This article summarizes the current semester's student experiences and suggests possible scope for improvements.

1 Introduction

Course-based undergraduate research experiences (CUREs) offer an effective way of integrating research into an undergraduate science curriculum and extending research experiences to a large, diverse group of early-career students [2, 5]. Students who participate in CUREs develop content knowledge and technical skills specific to the area of research [8]. An increasing number of well-designed and well-controlled studies show that CUREs can influence

students learning, development, and educational and career trajectory[3, 4, 7]. Several studies [2, 5] discuss about research-focused courses in different STEM fields: Biology, Chemistry, Medicine, etc. In the Computer Science (CS) discipline, though there have been some practice already in different organizations, a model for such course has not yet been reported to the academic community. This article details a model of a CS course following the principles of course-based undergraduate research experience.

In this paper, section 2 describes the course structure and activities, section 3 states student experiences and outcomes, section 4 makes suggestions about extending the goals and involvements beyond the classroom, and section 5 draws conclusion. Due to the page-limit constraints, detail definition and logical structure of CURE, short and long term benefits of such courses are omitted. Interested readers might find relevant information in article [2].

2 CURE in Practice for CS Students

In Computer Science, research areas such as Natural Language Processing, Information Retrieval, Machine Learning, Artificial Intelligence, Data Science, etc. do not always need specialized equipments/hardwares other than a modern computer-equipped lab space (teaching-lab) for deploying a full research-focused course. The experience report is based on such course in Information Retrieval(IR) track. The next couple of sections outline the structure and activities of the course.

2.1 Course Layout

The course objective is to provide hands-on experiences with several existing software tools through the programming assignments and to train them to develop critical thinking and analytical skills through written assignments. In order to offer a balanced way of learning new techniques and applying them for conducting research, the course is divided into two main phases:

- Supervised Learning Phase
- Semi-supervised Learning (Research) Phase

The next two sections introduce the major activities during these phases.

2.2 Supervised Learning Phase (Week 1 $\sim$ Week 7)

The activities of this phase are mostly **lectures** highlighting and demonstrating important concepts from textbook readings. When appropriate, external readings can be used. Students are asked to complete written assignments (individual works), in-class tasks and programming assignments (group works).

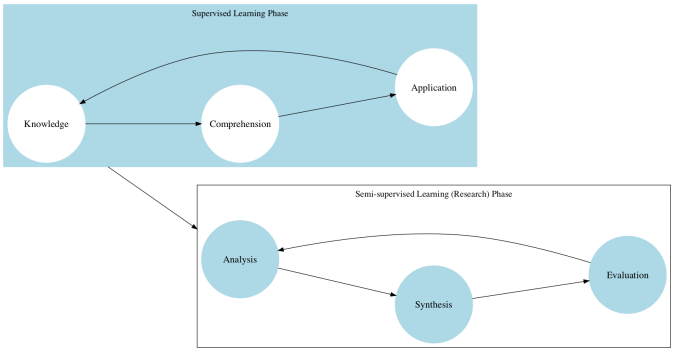

Figure 1: A revised Bloom's Taxonomy [1] for a research-focused course.

2.2.1 In-Class Tasks

Providing students with models and worked examples can help them learn to solve problems faster [6]. During the second hour of every class, students either run/test small experiments (with code provided by the instructor) or implement/modify a partially solved model to get a better understanding.

2.2.2 Programming Assignment

The programming assignments are designed with the following goals in mind:

- Students should have a lot of freedom in choosing the layout and the techniques while implementing the solution.
- Students should know about the standard evaluation techniques and be able to design new scales of measures.
- Students should write detailed reports for the experiments.
- Students should be comfortable using different standard datasets so that they can test their work with larger collections.

2.2.3 Written Assignment

Students are asked to submit written reports (summarize the concepts and answer few related questions) about published articles (chosen by the instructor) during the first and the fourth week. The goal is to improve their analytical and writing skills on technical matters.

Programming assignments and the template for written assignments can be found in appendix A.

2.2.4 Term Exam

The supervised learning phase ends with an open-book, two-hour long exam where students are asked to solve analytical questions on covered topics.

The covered topics on this phase can be found in appendix B.

2.3 Research Phase (Week 8 $\sim$ Week 14)

The first day of this phase is a full lecture led by the instructor introducing the students to different research sub-areas and possible research problems in each of those areas. At the end of the session, students form groups on their own and start talking about possible research problems that sound interesting to them.

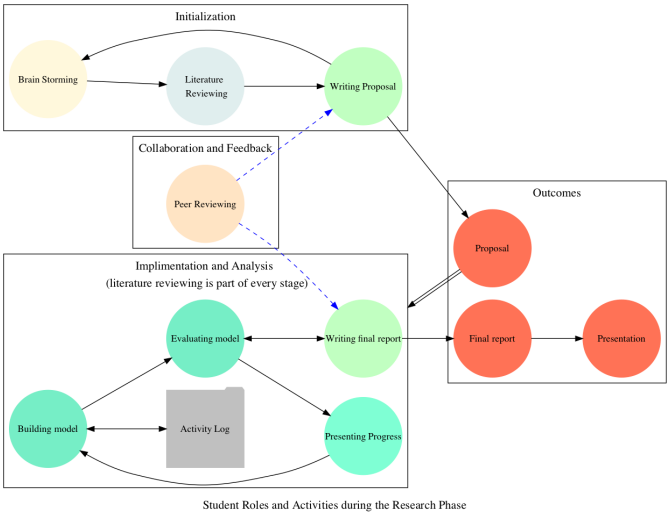

Figure 2: Major Student Activities during the Research Phase

The instructor takes a passive role (monitor and guide) from now on. During the eighth week, students spend their time on writing research proposals which are approved after two trials of presentation (one informal and one formal) in front of the class.

As **literature reviewing** is an important part of research, each student-group presents two articles (relevant to their research idea) as a basis of their work. They also submit a weekly plan: approximate amount of spent time per week and tentative weekly goal. Asking them to budget time is important as this helps them find a balance between their ambitions and achievable

goals within time and other constraints. Based on the demand of the research project, students may need to find relevant datasets or state a plan for creating datasets.

Template/Guideline for Writing A Research Proposal

Title: Find an interesting and representative title.

Abstract: Highlight the basic idea and its importance.

Project Description: With few literature reviews, explain your idea, the scope, the broader impact of your project.

Datasets: Cite proper existing datasets (if required) that you plan to use or explain how you plan to construct one for your work.

Preparation: Report the skills and knowledge that you think will be useful.

Outcomes & Timeline: Highlight some final expected outcomes (a paper, poster, software, API, etc.)

References: Cite relevant research works (at least 5 to begin with).

To make the students accountable for maintaining the quality of their work, every student also work as **anonymous reviewer** for other students' projects. The final work is submitted as a paper following standard formats commonly found in most technical IR articles: title, abstract, introduction, related work, dataset, methodology, result analysis, and conclusion/future directions.

Review Template

1. Summarize the project-idea in your own words.
2. Rate at the scale of 1 to 10 (1 being the worst and 10 being the best):
    - Is the research question clearly stated?
    - Have they explained clearly why their idea is interesting and impactful to some parties?
3. State some strengths and some weaknesses of the project idea.
4. Have they clearly talked about the evaluation mechanism and datasets (if needed) for their project?
5. (Optional) Provide some suggestions to the team.

Careful attention is paid on their **writing quality**: explaining the methodology and contribution, citing relevant works, acknowledging peers for their comments and suggestions, etc. The 14-week long activity ends with formal 20-min presentation by each group in a seminar hall where students from the school will be invited to attend.

2.4 Time Commitment

The students meet twice a week for 2 hours in a teaching lab. It is expected from the students that they would complete assigned readings (approximately 1∼2 hours) for the class and complete the assignments outside of class-meetings (approximately 4∼6 hours). Overall, students are expected to spend approximately 7∼10 hours per week for the course.

2.5 Grading/Assessment

50% of the course grade is allocated for the research phase. Clarity of thinking, efficiency of implementation, depth of analysis, structure and readability of the project report, and final presentations are the determining factors for achieving that 50% score. The rest of the course-grade is distributed on regular programming assignments, written assignments, class participation, and one term exam. A complete course syllabus and related materials can be found at authors github repository: `https://github.com/FahmidaHamid/CCSC2020CP`

3 Students Experiences

Given the research sub-areas as {Keyphrase extraction and Summarization, Learning to rank, Question-Answering, Recommender Systems}, the running student projects are the followings:

Student Projects

Information Desk: to create a powerful Question-Answering system, leveraging the power of Generative Adversarial Networks (GAN) and the World Wide Web (WWW).

Teaching Machines: to Reason on texts, to build a Question-Answering model with complex questions (how or why) and eventually understand the sequential facts embedded in the documents to produce the answer.

Keyphrase Extraction of YouTube Video Transcripts: to outline the contents and to create tags to improve search results.

A Study of Keyphrase Extraction on Celebrity Tweets: to create a celebrity-tweet dataset and apply unsupervised keyphrase extraction techniques on it.

The first two groups are studying several Machine Learning techniques for implementing automatic answer generation in different situations. The

third group is using YouTube's API for building their own "transcript dataset" and applying meta-information to build a Naive-Bayes Classifier for extracting meaningful phrases. The fourth group has manually created a dataset of "Monthly Celebrity-tweets" and will be using an unsupervised technique to automatically extract keywords. The first and the third group will use human annotators to help create gold-standards for evaluating their system's performances. The second group uses a dataset published by Facebook that comes with standard answers. The group plans to extend the model from producing short phrased answers to a complete sentence.

The major outcomes of the course from the student perspectives are:

(a) To be able to write a research proposal and a technical paper.
(b) To be able to think independently and discover a problem (area of concern)
(c) To be able to write constructive criticisms of other's work.
(d) To plan for publishing the work as a student paper/poster to a proper venue.

4 Extending the Scope and Possibilities

If planned with enough time and resources, several other possibilities can be included to add more value and learning experience:

Invited talks: Scholars in similar research tracks can occasionally give invited-talks. In some cases, students can remotely join at different events (e.g. ACM TechTalk) and thus flip a lecture on similar/same topic.

Student talks: Offering bonus-points for actively participating in some student clubs and presenting the research will help students build confidence and acquire more knowledge. This can motivate the students in audience seats for conducting research as well.

Workshop/Poster presentations: Students should aim for presenting their work to external events. Regional or national undergraduate research symposiums are possible suits for their work."What will be the impact of my research?", "Why is this problem still unsolved?" – these questions lead students into determining the scale of their work and not limit their achievements only to a good letter grade.

5 Conclusion

The goal of the course is to lead the students to the process of scientific discovery in a small scale. The first round of experiment at Grinnell College with eight senior students has been successful in several ways. Given more time

and scope, in next trial, several modifications can be done: having an overlapping research and supervised learning phase, offering student-stipends for teams that can participate at poster/workshop events outside of the college, etc. In short, research-focused courses create a trend of involving undergraduates into research in an effective way. Faculties with different research interests should offer research-focused courses by rotation to engage larger and diverse set of students.

References

[1] Nancy E Adams. Bloom's taxonomy of cognitive learning objectives. *Journal of the Medical Library Association: JMLA*, 103(3):152, 2015.

[2] Lisa Corwin Auchincloss, Sandra L. Laursen, Janet L. Branchaw, Kevin Eagan, Mark Graham, David I. Hanauer, Gwendolyn Lawrie, Colleen M. McLinn, Nancy Pelaez, Susan Rowland, Marcy Towns, Nancy M. Trautmann, Pratibha Varma-Nelson, Timothy J. Weston, and Erin L. Dolan. Assessment of course-based undergraduate research experiences: A meeting report. *CBE: Life Sciences Education*, 13(1):29–40, 2014.

[3] M Kevin Eagan Jr, Sylvia Hurtado, Mitchell J Chang, Gina A Garcia, Felisha A Herrera, and Juan C Garibay. Making a difference in science education: the impact of undergraduate research programs. *American educational research journal*, 50(4):683–713, 2013.

[4] Sylvia Hurtado, Nolan L Cabrera, Monica H Lin, Lucy Arellano, and Lorelle L Espinosa. Diversifying science: Underrepresented student experiences in structured research programs. *Research in Higher Education*, 50(2):189–214, 2009.

[5] Jane L. Indorf, Joanna Weremijewicz, David P. Janos, and Michael S. Gaines. Adding authenticity to inquiry in a first-year, research-based, biology laboratory course. *CBE: Life Sciences Education*, 18(3), 2019. PMID: 31418655.

[6] Barak Rosenshine. Principles of instruction: Research-based strategies that all teachers should know. *American educator*, 36(1):12, 2012.

[7] P Wesley Schultz, Paul R Hernandez, Anna Woodcock, Mica Estrada, Randie C Chance, Maria Aguilar, and Richard T Serpe. Patching the pipeline: Reducing educational disparities in the sciences through minority training programs. *Educational evaluation and policy analysis*, 33(1):95–114, 2011.

[8] Jack T. H. Wang. Course-based undergraduate research experiences in molecular biosciences-patterns, trends, and faculty support. *FEMS Microbiology Letters*, 364(15), 07 2017.

A Sample Assignments

Programming Assignment 01
Allocated Time: 2 weeks
Goal: To implement one of the most common applications in IR
Problem Statement:
Build a search engine using your preferred programming language. Your code should be well documented. While implementing your engine, follow the basic building blocks of a standard search engine. Report the strengths and weaknesses of your model.

Server Side should do the followings: crawl at least 100 unique pages, parse them, create index, build web-graph by analyzing the links, and rank them (you may want to implement the page-rank algorithm). It is a good idea to design a topical search engine, like, crawl pages whose title contains a particular topic (e.g. sports). That way, your web-graph will be well-connected. Client Side should have a very simple interface to take input (text query) from the user, search in the index, and print the related web-links in some order. Bonus points will be allocated for designing interactive user-interfaces.

Programming Assignment 02
Allocated Time: 2 weeks
Goal: To find the latent features from the so-called unstructured and abnormally large collection of data.
Problem Statement:
The spam filtering problem is one kind of text categorization problem with the categories being spam and ham. The structure of email is richer than that of flat text, with meta-level features such as the fields found in MIME compliant messages. Researchers have recently acknowledged this, setting the problem in a semi-structured document classification framework. Several solutions have been proposed to overcome the spam problem. Among the proposed methods, much interest has focused on the machine learning techniques in spam filtering. They include rule learning, Naive Bayes, decision trees, support vector machines, or combinations of different learners. The basic and common concept of these inductive approaches is that using a classifier to filter out spam and the classifier is learned from training

data rather than constructed by hand.

Your job is to choose two possible machine learning approaches to solve the problem and analyze their performances. Then you will report possible mechanisms to improve the current models you implemented.

Written Assignment
Allocated Time: 1 week
Study a given article “ABC" and answer the following questions:

- Summarize the article. (State the overall contribution of the article within 500 words.)
- Explain their hypothesis using the following structure: (assume a relevant structure M is given.)
- The authors used mechanism P and Q for evaluation. Suggest another relevant evaluation mechanism.
- State at least two cases where the stated hypothesis will fail (or, State some weaknesses).
- Find another problem where you think the hypothesis (solution) can be applied equally effectively.

B Covered Topics

Topics Covered
Boolean Retrieval Model
Search Engine Architecture
Web Crawler & Basic Text Processing Technique
Inverted Index & Query Processing
Search Result Interface & Link Analysis
Vector Space model
Probabilistic Information Retrieval
Language Models
Evaluation in IR
Discussion on possible Research Tracks
- Question-Answering
- Recommender Systems
- Keyphrase Extraction
- Learning to Rank