SI 699-002 - Search and Recommender Systems

Mastery Course

Fall 2017

Instructor: Kevyn Collins-Thompson

Classroom: Tues 1-4pm NQ 1265

Instructor: Kevyn Collins-Thompson

Email: kevynct@umich.edu

Office: NQ 3344

Office hours: 4-5pm Tues or by appointment

If you have questions about the course or your project, please feel free to come and talk with me during my office hours. You can also contact me via email: please put “699” in the subject line so I can be sure to attend to it. (Please note that I may not be available on email over the weekend.)

Course Email: via Canvas

Note: Some syllabus details may be subject to change.

This is a project-based course: there is no midterm or final exam for this course. However, there are milestones/progress reports in addition to the main project that must be completed satisfactorily to pass the course.

Overview

Have you ever wanted to build your own search engine? Or do you have ideas for how to make current search engines or recommender systems better, or useful for a much more diverse set of users or tasks? Or perhaps you've always wanted to explore how to support new technology for connecting people with the information they're looking for - conversational chatbots, voice recognition (Amazon Echo/Google Home), augmented reality interfaces, or motion and gesture sensing, for search?

The search and recommender systems mastery course offers scope for exploring your ideas about better ways to build and evaluate core search and recommendation technology. The course will require students to demonstrate mastery of theoretical models, algorithms, evaluation methods, and user interfaces for search and recommender systems. Students will synthesize methods from information retrieval,
natural language processing, machine learning, HCI, and related fields. Students will work on semester-long projects that address real-world problems. Aligned with best industry practices, students will be expected to work in a fast-paced environment and to demonstrate independence and creativity as well as technical mastery.

By taking this course, students are expected to:

1. Identify a real world scenario that requires a search or recommender system problem to be solved, by identifying the input (user information need, content, and their representations, the output (from the search/recommender system), and approach needed to generate the desired output.
2. Formulate the problem (input -> output) using information retrieval or recommender system paradigms, including but not limited to user and content representation (features), indexing, classification, clustering, ranking, prediction, or when appropriate, user interface design.
3. When appropriate, identify state-of-the-art algorithms and methods for addressing a search/recommender systems task.
4. Collect or index the content needed for the retrieval task/system.
5. Collect training data needed for any predictive algorithms.
6. Design experiments to evaluate/judge/measure the success of the search/recommender system.
7. When appropriate, deal with content and indexes at scale (by writing programs and using correct tools).
8. Use statistical and visualization tools to describe the properties of the system.
9. Validate, summarize, and present an evaluation of your system.
10. Draw correct conclusions from your evaluation.
11. Disseminate your results to the broader community in the format of poster presentations, social media posts, websites, etc.

Prior to taking this course, students are expected to have competency in programming, data manipulation, statistical data analysis, and basic information retrieval concepts, paradigms, and evaluation methods. Depending on the computing environment, you may also need to know or learn how to work with Unix environments, and/or how to configure and using state-of-the-art data mining and statistical analysis tools.

Students should also be, or become, familiar with common practices of managing individual source code projects, such as version control (e.g., git), project documentation (e.g., wiki), and progress tracking (e.g., trello).
As a prerequisite to this course, students must have completed the following MSI courses:

- SI501
- SI506/507
- SI544 (statistics and data analysis, or equivalent)
- SI561 (natural language processing)
- SI650 (information retrieval)

and at least two of the following:

- SI618 (data manipulation and data exploration, or equivalent)
- SI608 (networks)
- SI671 (data mining)
- SI649 (information visualization)

or obtained a waiver for a specific requirement from the instructor.

Course Materials

There are no required texts for this class. Any readings will be provided online as needed. Any lecture slides will be made available on Canvas before and after each class meeting.

There is an excellent supplementary text that's recommended highly:


Abstract: [https://pdfs.semanticscholar.org/1bda/16bf151b501626e186dcbcd341ace858e238.pdf](https://pdfs.semanticscholar.org/1bda/16bf151b501626e186dcbcd341ace858e238.pdf)


Grading (may be subject to adjustment)

- 60% on final project
  - Degree of meeting initial commitments (e.g. bare minimum, good, excellent) 40%
  - Final report 10%
  - Final poster presentation 10%
  - Extra credit may be awarded for especially creative and/or impressive projects.
- 25% on midterm progress report and presentation
- 15% on other project milestones/progress reports
• Each project must satisfactorily complete all evaluated items, including the final report and presentation – regardless of grade – to pass the course.
• Regular participation in class sections, when meeting, is required.
• In class meetings, each student will give a brief presentation on progress via an online wiki and summarize in class.

**Late days.** In the case where there are graded items to be handed in, you have 3 penalty-free late days. One late day = one 24hr period after due date. No fractional late days: they’re all or nothing. 25% penalty per day after late days are used up. You don’t need to explain late days - we track them for you. Submit late assignments via course site as usual. Late days can't be applied to the final project deadline.

**Course Project Schedule**

The instructor will provide a few sample problems and large-scale data sets. Students are encouraged to propose their own problems and data sets, which needs to be approved by the instructor before Week 3. In Fall 2017, example of data sets may include (1) The HathiTrust e-book archive and index (millions of books and billions of words); (2) CommonCrawl web snapshot across billions of web pages.

Projects are individual. Each student should meet with the instructor to discuss project progress (which may or may not be scheduled during the class sessions). Feedback on project progress will be updated on Trello and documented in the course Wiki. In-class sessions will be used to discuss issues that are related to all students (such as tutorials of tools and algorithms). In particular, the first two weeks of class will be dedicated to a customized introduction to computational environments, evaluation methods, datasets, and sample projects. All projects will be finalized by week 3, with each project presenting a project proposal in front of the class. A mid-term project presentation will be held in class in week 7 or 8, and a final public presentation of the projects will be held in the last week of class. Depending on In other weeks, teams do not need to attend in-class sessions unless for weekly progress report meetings. Each student must show measurable progress each week.

Every student will be assigned to a different project so that there is no direct competition between projects. Project documentation will be kept up-to-date on the course Wiki and be shared among everyone in the class. Students are encouraged to make constructive
peer critiques during the opening presentations and mid-term presentations and to provide suggestions and help to each other throughout the semester.

In class sessions, every student will give a brief update on their progress.

A rough schedule of the project milestones is as follows:

- Week 1: Overview of the course, introduction to sample projects and datasets.
- Week 2: Project selection
- Week 3: Project design finalized and presented in class
- Week 8: Mid-term report submitted, reviewed, and presented in class
- Week 13: Project results submitted and reviewed.
- Week 14: Final presentations and result dissemination

Specific calendar follows (may evolve as needed).

START: Tuesday, Sept. 5, 2017 (1-4pm).
- 1-2pm: Weekly team updates and discussion
- 2-4pm: Instructor/project meetings

Tues. Sept. 12: Background topics, discuss potential projects.


Tues. Sept. 26: Progress report 1. Initial project plan due

Tues. Oct. 3

Tues. Oct. 10: Progress report 2 due


Tues. Oct. 24: Midterm checkpoint and presentation due

Tues. Nov. 1

Tues. Nov. 7: Progress report 3 due

Tues. Nov 14

Tues. Nov 21: Progress report 4, w/ report update due

Tues. Nov 28

Tues. Dec 5: Progress report 5 due
Tues. Dec. 12:

- Final project poster presentation
- Final project report

**Academic Integrity and Original Work Policy**

Unless otherwise specified in an assignment, all submitted work must be your own, original work. You may discuss general approaches with others on individual work items or project components, but you may not copy code or other work wholesale. If more than one person contributed to a graded item, you must indicate on your turned-in work who you worked with, and how you worked with them.

Any excerpts from the work of others must be clearly identified as a quotation, and a proper citation provided. Any violation of the School's policy on Academic and Professional Integrity will result in severe penalties, which might range from: failing an assignment, failing the course, or being expelled from the program, at discretion of the instructor & Assoc Dean for Academic Affairs.

**Accommodations for Students with Disabilities**

If you think you need an accommodation for a disability, please let me know at your earliest convenience. Some aspects of this course, e.g. assignments, the in-class activities, and the way we teach may be modified to facilitate your participation and progress. As soon as you make me aware of your needs, we can work with the Office of Services for Students with Disabilities (SSD) to help us determine appropriate accommodations. SSD (734-763-3000; [http://ssd.umich.edu/](http://ssd.umich.edu/)) typically recommends accommodations through a Verified Individualized Services and Accommodations (VISA) form. I will treat any information you provide in as confidential a manner as possible.

**Student Mental Health and Wellbeing**

The University of Michigan is committed to advancing the mental health and wellbeing of its students. If you or someone you know is feeling overwhelmed, depressed, and/or in need of support, services are available. For help, contact Counseling and Psychological Services (CAPS) at (734) 764-8312 and [https://caps.umich.edu/](https://caps.umich.edu/) during and after hours, on weekends and holidays, or through its counselors physically located in schools on both North and Central Campus. You may also consult University Health
Service (UHS) at (734) 764-8320 and https://www.uhs.umich.edu/mentalhealthsvcs, or for alcohol or drug concerns, see www.uhs.umich.edu/aodresources.

For a listing of other mental health resources available on/off campus, visit http://umich.edu/~mhealth/