

605.744 Information Retrieval: Class Project

Goals

The class project allows you the opportunity to investigate a particular topic of information retrieval in greater depth than we could cover in the classroom. Projects are usually individual endeavors, and require advance coordination and continual effort to complete successfully. There are two types of projects: ones oriented around an empirical investigation (those that build an interesting software system or conduct an experiment), and those primarily analytical or non-quantitative in nature. In previous classes the majority of projects have been empirically based and involve writing or using software to conduct research. For example, you might use a publicly available IR engine to conduct an experiment using a test collection, you might write a web spider to build a collection of web documents so you can analyze the graph structure or properties of hypertext collections, or, you could develop a new algorithm for text classification and compare it against a known baseline. For such experiments, the relevant literature must be briefly reviewed to give context to your work, a hypothesis must be developed, experiments must be designed, instrumented, conducted, and analyzed, and finally, results must be presented to the class (and a short written report must be submitted, around 5 pages in length). The other type of project will require a much more extensive review and study of the literature and must exhibit independent thinking; the written report for theory papers will be scrutinized in greater detail, and is expected to be of greater length (~ 15 pages) and of superior quality. Analytical papers may focus on a research idea or on a solution to an applied problem.

Grading Criteria

Experimental projects involve appreciable software use, or software development, and will be graded differently than theory projects:

Empirical Study

10% proposal
20% experimentation
20% analysis
30% written report (≥ 5 pages)
20% in-class oral presentation

Theoretical Study

10% proposal
70% written report (15 - 20 pages)
20% in-class oral presentation

Generally, it will be harder to get a low grade for a decent experimental paper, though it is possible. However, theory papers must be of high quality to receive a grade of 'A'; poor ones, those hastily researched and written, will receive a low score ('C' or below). An 'A+' (rarely seen) indicates that with additional work and revision, the paper might be submittable to a research conference.

Proposal

A written proposal must be submitted and approved. The proposal should identify a topic of interest, briefly motivate why this is an important problem, state the type of the project (experimental or analytical), identify some relevant literature for the problem of interest, briefly compare the proposed work to the available literature, and outline planned work for the project. For an experimental project, sufficient detail into the experimental design must be given; for a theory paper, slightly more discussion of the literature is expected in the proposal. Proposals should be 1

to 2 pages in length (no more). If you have a topic that interests you, but you aren't sure how to proceed, you are welcome to contact me for ideas, even before the proposal is due.

Written Report

Reports should be scientifically-oriented and ought to include an abstract, an introduction to the problem being considered, a review of related work (extensive if a theory paper), discussion of ideas (extensive if a theory paper), experimental results with analysis (extensive if an empirical paper), conclusions supported by your work, and appropriate references. To compute page lengths you may assume 500 words per printed page. Generally the style of formatting is up to you; however, do include headings, and use a font of 10-12 points. Suitable tables and figures are welcome. Written reports should be handed in during the last class meeting.

Presentation

An oral presentation must be delivered to the class near the end of the semester. Your talk should be 15-20 minutes in length and should model a conference presentation. Students should use prepared overheads or electronic (*e.g.*, PowerPoint) slides. Presenters should clearly introduce the problem under discussion, review prior work from the literature, explain in detail your contribution and results, and be able to intelligibly field questions. Experiments are not always successful, you can achieve a good score for this part of the assignment, even with 'negative' results; however, your design must be good and you need to articulate what *was* learned. If your work is theoretical you should illustrate how it can be evaluated and what applications should benefit from your ideas.

Schedule

3/5-3/12	Choose a topic and submit a proposal (one page can be enough; email is OK)
4/9	Status report (one paragraph can be enough)
4/30	Student Presentations
5/7	Student Presentations

Literature

Numerous resources are available to you. Online papers can be found via Google, CiteSeer, or various websites for conferences; pointers to the TREC conferences and the ACL Anthology are on the course web page. I believe that the JHU libraries can provide access to the ACM and IEEE digital libraries. Finally, I may be able help in obtain copies of difficult to find papers using my private collection.

Sample topics by former students

- Evaluating indexing and retrieval of Hindi song titles
- Exploring methods to compress indexes using document identifier reassignment
- A prototype system for indexing blogs
- Extracting obituary information from news sources for genealogical purposes
- Distributed indexing and Bloom filters
- Predict author gender, time of authorship, or identify of author
- Extraction of apartment rental information from Craigslist ads
- Proposing a model for click-through fraud

Theory / Application Ideas

- Why hasn't word-sense disambiguation been able to improve IR performance?
- What problems are current large-scale evaluations (like TREC) susceptible to?
- How can effective information access be achieved on mobile devices?
- Develop a framework for retrieval against scanned documents
- Investigate retrieval of a specialized type of document (*e.g.*, a collection of source code or job openings).
- How can the relative efficacy of two *web* search engines be established?
- How can spam filtering software adapt to new changes (trends) from commercial spammers?
- How can commercial engines counter attempts to falsify click-throughs?
- Extensively compare desktop search tools (Google's Desktop Search vs. Apple's Spotlight)
- Could statistical language models be employed to ascertain author identification?
- Can IR techniques be applied to discover information about protein or genetic sequences?

Empirical Ideas

- Can POS-tagging be used to improve IR performance?
- Can a given NLP technique improve performance (*e.g.*, keyword phrases or stemming)?
- Investigating different methods for tokenization (and their effect on retrieval performance)
- Using electronic thesauri to automatically augment user queries
- Learn to spell correct or phrasify (add quotes to) user's web queries.
- Apply a machine learning algorithm (*e.g.*, SVMs/NNs/Decision Trees) for text filtering
- Develop and test a method for spam filtering.
- Implement an algorithm for document similarity that we did not cover extensively in class, such as Cover Density Ranking or Latent-Semantic Indexing. Compare your results to some baseline method such as vector-cosine or an out-of-the-box IR package.
- Experiment with cross-language retrieval by translating (by hand) some of the TIME queries into another language and using a bilingual dictionary or on-line MT system to translated queries back to English prior to search.
- Write a mini intranet search engine with at least 500 documents, analyze the contents and demonstrate an engine to search it
- Implement phrase-indexing efficiently (see work by Bahle et al.)
- Build and evaluate a translation resource (*e.g.*, dictionary or parallel corpus) obtained from the Web
- Obtain a speech recognition package and run it on web-available audio files to support retrieval. (see <http://www.ece.msstate.edu/research/isip/projects/speech/>).
- Help users visualize textual information (such as a retrieved document set)
- Develop a system for retrieval of music
- Build a collection from the Web of 50k documents. Analyze it in depth.
- Attempt retrieval of stored images
- Develop an information extraction system that learns a particular kind of fact from unstructured documents (*e.g.*, crimes: perpetrator, victim, date, officers involved)
- Build a system for collaborative filtering, to match people with similar interests, or to suggest movies or books to an individual based comparing their profile with other's
- Analyze an atypical corpus. For example, an email archive of Enron documents is available.
- Question answering for one particular type or question (*e.g.*, how many or who).