WEB SCIENCE -- FALL 2011
PROJECT 1: Building a "better" search engine.

Deadlines:
Project assigned: September 23, 2011
Groups declared on Wiki: 11:59 PM, September 27, 2011
TGIF Presentation: October 7, 2011 (in class)
Final submission of Paper and (reviewed) Code:
11:59 PM Tuesday, October 11 (Submission details to be provided later)
Class on September 30 will be a lab session for this project.

(Please note – Tues Oct 11 is a Monday schedule day, so no class).

You may work in groups of ***three or four*** (not less, not more) for this assignment. You may pick your own groups, but they need to be decided, named, and published on the wiki, before midnight on Sept 27.

Project

For this project you will create a specialized search engine that biases its results to help promote a cause of your team's choice. Your team can choose a cause that is real (for example, helping the victims of the famine in the Horn of Africa) or far-fetched (such as, saving the cows of Earth from being abducted by aliens). Remember that you will be presenting your work to the class, so try to avoid embarrassing yourselves or the Professor by picking something non-offensive. (If you're not sure, remember the Google ask the Professor).

Using a powerful search API that we will provide, you will be able to get a set of good page results, up to 200, for a search query (all will be relevant to a query term) – so you won't have to build the search engine. However, to promote the cause you to do two things: First, you will need to sort the results of the query to see if any pages related to your topic can be found, and if so, to promote them to the top of this search results page and annotate them with an ad for your cause. Second, you can manipulate the query (within reason, see below) to improve the chances of finding the relevant term.

So let's say you chose the example above, protecting the cows of Earth. Suppose the query that came in was "Rensselaer Polytechnic Institute" and somewhere in the 200 was the campus dining hall. Rather than the normal result, which is:

> **Campus Dining - *Rensselaer Polytechnic Institute (RPI)***
> *rpi.edu/student_life/dining.html* - **Cached** - **Block all rpi.edu results**
>
> Dec 17, 2010 – About *RPI*, Academics, Research, Student Life, Admissions, News, Tour **...**
> Each dining hall offers a variety of *food* formats that can be enjoyed **...**

you could use "dining" as a signal that would allow you to change that into a result such as

Campus Dining - *Rensselaer Polytechnic Institute (RPI)*
rpi.edu/student_life/dining.html **- Cached - Block all rpi.edu results**

Dec 17, 2010 – About *RPI*, Academics, Research, Student Life, Admissions, News, Tour **...**
Each dining hall offers a variety of *food* formats that can be enjoyed **...**
[[IF YOU ARE EATING MEAT BEWARE, ALIENS ARE STEALING OUR COWS]]

and to make sure it showed up near the top of the search results (any results you alter should show up before the remaining searches).

However, for most random queries, you will not necessarily be able to find something particularly relevant. Thus, you will need to do something to make it more likely that your cause will be promoted. For example, if the user typed the query "Web Science" there would not likely be too much of interest about saving the cows. If, however, you were to change this to "Web Science on Mars" there'd be a lot more likely to get one of your taglines.

However, here's the catch, you must somehow make it so the user would not be likely to guess the manipulation you did (otherwise they would obviously never use your search engine again). This will take a combination of good tactics in your design and cleverness with respect as to how you do it.

Search Engine
Your search engine will be built by extending the Google search engine. We will provide you with an API for this (see below). You will develop (and document) search heuristics to rearrange the results so they are optimized for your domain. What heuristics and strategies you use to guide this rearranging of your results is all up to you. But please remember: Don't be evil.

Your search engine must be written in Python. You will be using a modified version of the python xgoogle library by Peteris Krumins found at http://www.catonmat.net/blog/python-library-for-google-search/

You will not be using the exactly library found here, but one we have modified which will be found on the class wiki. We will give you an abstract class that uses this library to query Google and returns the search results. You will realize this abstract class with your code to manipulate the query and/or sort the results. (Examples of the use of the library will be made available on the class wiki).

When you submit your code you should note that

1) Everyone on your team needs to have looked at it to make sure you all know how it works. I reserve the right to ask any team member to explain not just the approach, but how it is built.
2) Someone from another team must review your code. As part of the submission of you paper, you will include the name of your reviewer. The reviewer will get extra credit for their doing the extra work. Note, however, if the reviewer does not detect a major code bug or otherwise misses something they shouldn't have, they will not get the points.

As this is the first time we are doing this, we will keep the reviewing informal. Remember, the goal of the review is to make sure the code works, to look for ways to help make it better, and to help the team you are reviewing get full points for their code.

You will submit your source code for this part of the assignment.
The working search engine embodying the strategies developed by your team is worth 60% of the grade on this project.

Project Report
For this project you will also write a report on your prototype, approximately 5 pages. It should cover the strategies and heuristics you used for making your search engine. You must explain why you felt these strategies and heuristics would work for your cause and evaluate (with examples) what did ***and didn't*** work and what you would do in the future given the time and resource to "do it right". You are encouraged to research ideas from other papers, blogs, etc but, of course, you must reference them in your paper. You are particularly encouraged to bring in ideas from the reading or from topics discussed in class that help justify your approach and results.
This paper will be 30% of your grade on this project. This is where the points for the "cleverness" of your solution will be evaluated.

TGIF – Oct 7
You will prepare a 5-minute TGIF presentation about your project for the class. This presentation can be made by one or more members of the team. This presentation should explain how your system works, and show some examples of the expected (or implemented) behavior. You should be sure to explain as much of the "range" of your search engine as you can. Due to time constraints, we will not be able to have the sort of heavy-duty Q/A that Google TGIFs include, but do expect to be asked some questions about your work (all team members)
The presentation/demo will be worth 10% of your grade

Extra Points:
1) A set of evaluators, including the professor, will try all the search engines and pick the best one. All members of the team submitting that search engine will be awarded 5 extra points.
2) Anyone who reviews code for another team will receive up to 5 extra points (based on the quality of the reviewed code) but only once (i.e. there's no extra reward for doing multiple reviews).
3) Participation points count to your final grade, however particularly useful things put on the wiki may gain an extra point or two in reward. If you put useful code on the wiki, then you will get 2 points for each team that uses it and teams using other people's code correctly will receive 1 extra point.

Papers will be submitted electronically, details to be announced. Each paper submission must start with a "header" including:
Team Name.
Team Member Names.
Code reviewer Name.
Code reused details.
Brief description of the "cause"

Please note: This is not intended to be primarily a coding project, although there may be some clever coding needed. Note that the points for how good your ideas are make up a considerable amount of the grade. A team that includes a mix of talents is likely to have the best shot at doing well on this assignment. Note also that thinking hard about what you are trying to achieve,

experimenting and exploring, and developing a sound approach are crucial. Don't wait till the last minute (and try to have fun with this).

Academic Integrity policy for Project 1
This project is bound by the integrity policy of the class as expressed in the syllabus.