WEB MINING
FINAL
Due Date 07.06.2020

In this final project, you are free to choose your own topic. The following project ideas should give you a clue about what the extend of your project should be. **If you want to implement one of the following ideas, go ahead and do it.**

If you want to do a project which is **NOT listed** here, you should take my APPROVAL first. Please email your project proposal to me at gurhangunduz@mu.edu.tr for my approval.

You can have **2-person teams** to work on this project.(you do not have to). Each person in a team should clearly state what part of the project he/she worked on, in the report. I will also setup a one to one(team) zoom meeting around 15 minutes with at least %50 of the class for the presentation of your final project. In this meeting you will demonstrate your project and answer my questions regarding to it.

Everyone should upload their project files into DYS.

Project ideas;

- We know that social networks tend to become polarized around controversial topics in social media, for example political discourse on Twitter. It has been suggested that social media increase such polarization, by making it easy for people to communicate only with those who think like themselves. This is the phenomenon of so-called "echo chambers." Can you think of a way to find evidence for such a causal relation?

- Wikipedia's publicly-available data set provides a wealth of information about the structure and evolution of a dynamic socially-edited forum. (i) Possible research includes automated fact-checking and user authority measures (date of registration, number of posts, average size and longevity of posts, etc). (ii) Another avenue of investigation would be correlating IP addresses / usernames with changes to specific hot-topic issues. An automated method to detect suspicious editing / revisions could aid in the identification of biased or self-serving modifications leading to the identification of the offending individuals / organizations.

- Create a meta-search engine that finds potentially "embarrassing" personal material (photos, videos, text, etc) by mining various sources such as search engines, social network sites, photo and video sites, etc. This could play an educational service by highlighting the dangers of posting private information on the Web.

- Develop an application (eg for the Google Desktop, the Yahoo Desktop, Windows Desktop, browser extension, Mac Dashboard Widget, or mobile platform) that implements some (simplified/extended) version of the HITS algorithm. This would be a client-based solution for the query-time analysis.

- The economics of Google Ads are inducing (creating incentives for) a "pollution" of information on the Web. People create fake "original content" with popular query terms to attract traffic and make a profit through advertising. This is being done both manually (by underpaid hired writers) and with automatic text generation scripts. Can we devise techniques to clean the Web from such pollution? UPDATE: Google recently announced changes to its ranking algorithm to demote so-called content farms.

- Try to come up with a list of top-X sites frequented by spam harvesters. For example I created a gmail account to submit a script to CPAN, which posts the email of authors on their site. That account has quickly become a honeypot with thousands of spam messages. So clearly spammers harvest emails from CPAN. What other sites? What are the worst? You could write a crawler that automatically posts email addresses to sites it encounters (message boards, etc) and then monitors which sites generate the most spam.

- A learning method to classify an arbitrary Web page as blog or not blog, for crawling purposes.

- A text mining algorithm to find a huge set of triples (email address, name, address) from crawls of personal websites, blogs, bios, etc. and cross-reference with structured databases such as ip-to-zip converter, phone book, and other online resources.

- You can implement one of the two following crawling approaches into the Frontier of a given crawler (Crawler4j - https://github.com/yasserg/crawler4j).

    - The first crawling strategy is depth-first search. In depth-first search, the frontier acts like a last-in first-out stack. The elements are added to the stack one at a time. The one selected and taken off the frontier at any time is the last element that was added.

    - The second crawling strategy is the best-first crawler. In this strategy pages are visited in the order specified by the priority values in the frontier. The priority is specified based on an estimate of the value of the linked page. The estimate can be based on topological properties (e.g. the in-degree of the target page) or content properties (e.g. the similarity between a query keyword and the source page).

Example of previously done projects.

Usage Statistics of Robots Exclusion Standard
Mining for Blog communities
KidsCrawler
Using Page History to Rank Search Results
Web Topology of the Indiana University Domain

Additional Requirements

You should create you report using a MS WORD or equivalent editor. ZIP(RAR) your report and project into a file named as; your id+name+HW#(for example 06233025AhmetKorkmazHW3.rar).

Please Do NOT use a simple text editor to prepare your report. Use MS word or equivalent editor(since it will include screen shots and database diagrams). Convert it to PDF before sending it.

You should include the following information in your report:

1- Your name, your student id, and homework number.

2- Explain your project briefly. List the technologies that you have used. If you use a template, give the reference showing where you got it.

3- Explain how your program works by including screenshots and outputs.