**MSIS 615 Final Project A**

The final project A requires developing an automatic web crawler to collect data from a website of your choice, and perform analysis using Python analytics packages.

Grading of the final project A includes system functionalities and presentation.

**1. System Functionalities (15 points)**

**Requirements:**

In your project, your team should decide what data source to crawl from, what database schema to use, and what kind of analysis to perform. Note that:

1. Depending on the expected difficulty, novelty, and amount of work, different scores are assigned to each component. The actual score you can get depends on the actual quality, difficulty, novelty, and amount of your work. For example, in the database part below, SQLite is relatively easy to use and you are likely to get 2~4 pts. MS SQL Server requires more work and if you do well, you are likely to get 3~5 pts. The working code should be demonstrated in presentation to claim for credits.
2. You can implement any innovative ideas that are not listed below.
3. The total is 15 points by summing up maximum points for crawler, database, and analysis.

**Required modules:**

**1. Web Crawler (2~5 pt)**

Identify a website as your data source, and identify target data your team plans to collect. Note that you should aim to collect as much data as possible, even if you do not initially expect to use the data for analysis. This is because retroactive collection could be painful if you found that you are missing some needed data later on.

Here are some more specific requirements:

a. The amount of data should be reasonably large enough that a web crawler is necessary.

**Good Example**: collect all reviews (journal title, average durations, review reports, overall rating, outcome, motivation) from <https://scirev.org/reviews>



**Bad Example 1**: collect all reviews in the first page of the website

**Bad Example 2**: collect only the journal titles from the website

b. You can use python request package, BeautifulSoup package, Selenium package, other packages not covered in class, or combine some of them.

c. The Scirev.org website above can be used as data source (relatively easy website to crawl data from). You are also encouraged to choose other contexts and identify other data sources.

2. **Database** (2~5 pt)

Set up a database to store your data.

a. You can use any database including sqlite, MS SQL Server, MySQL, MongoDB etc. But note that Excel is not a database. Also note that sqlite does not support multi-threading and remote access (which means you’ll have to run one computer non-stopped to finish data collection).

b. A task table is not necessary, but good to have. Multi-threading (run multiple crawlers at the same time to speed up) and break-point resuming (When the crawler program is interrupted, you can resume from where it stopped rather than to start from scratch) requires the design of a task table.

c. Database normalization is recommended, but not necessary. For example, for the scientific review example, it is recommended to have a review table, a journal table, and a journal-review bridge table. However, you can choose to have one big table to store all the data (with redundant journal information). There is no penalty for not normalizing.

d. After data collection, you need to demonstrate collected data in database matches your expectation. For example, For the Scirev.org, there are about 40\*190 =7600 reviews in total. So you need to have about 7,600 rows in the review table.

3. **Analysis** (2~5 pt)

Based on collected data in database, perform some data analytics to obtain insights.

Some analysis ideas (you don’t have to do all, the depth and quality are more important):

a. Descriptive analysis (1~2 pt):

**Example:** For the top 100 books, identify journals with the highest/lowest overall ratings, identify top 10 journals with highest rating, journals with highest/lowest number of reviews, average time for first round reviews, journals with longest/shortest time of reviews, average review qualities, journals with the highest/lowest review qualities, etc.

How do these statistics differ by journal disciplines?

b. Visualization (1~2 pt)

**Example:** Visualize some descriptive statistics above using Python package (not Excel!)

c. Regression (2~3 pt)

**Example 1:** Does rejection tend to yield low ratings from researchers?

**Example** **2**: How do length of first round of review, total duration, number of reviews and review quality affect overall rating? How do the effects differ between accepted papers and rejected papers?

d. Sentiment analysis (3~4 pt)

**Example 1:** What is the distribution of sentiments over all texts submitted to the website? Are negative reviews significantly more than positive or neutral reviews?

**Example w:** How do the emotions expressed in motivation text correlate with paper outcome? Do authors with rejected paper tend to submit highly negative comments?

f. Other text mining analysis (1~4 pt)

What are the most frequent unigrams, bigrams, and trigrams used in review texts, after removing stop-words?

What are the major topics discussed in the reviews (LDA analysis)?

**2. Presentation and submission of project presentation slides (15 points)**

Presentation Due Time: **shown on Blackboard, under the project submission link**

Files need to be uploaded:

1. Source code
   1. only upload.py files. Do not upload image or audio data. Upload to Blackboard project submission link
2. Presentation slides
   1. Summarize what data source you crawled, how you managed the data storage, and what analysis has been done. Upload to Blackboard project submission link

The following shows a rubric for grading students’ presentations.

|  |  |  |
| --- | --- | --- |
| **Rubric Item** | **Explanation** | **Point** |
| **Presentation** |  |  |
| Submission | Presentation slides and source codes must be submitted to Blackboard on time. Late submissions or submissions to wrong destinations receive 0 on this item. | 2 |
|  |  |  |
| Group | Students should report the group members by the deadline noted in syllabus. The number of students must be 2 or 3. Groups with more than 3 members, or groups which changed members after reporting will receive 0 on this item. | 3 |
| Formality | Presentation slides are typo-free and grammatical error-free.  Points will be deducted depending on the amount of typos and errors. | 3 |
| Time Control | Presentation length should be between 10-15 minutes   * If it exceeds 15 minutes, it will be stopped immediately and this item receives 1.5 points If it is less between 8 ~ 10 minutes, this item receives 1 point * If it is less between 8 minutes, this item receives 0 point | 2 |
| Visual | The presentation slides should be visually pleasing. Students need to carefully choose font size, font color, background color, amount of texts in a single slide, good blending of texts and pictures, placement of pictures, figures and tables | 2 |
|  |  |  |
| Communication | Students should present their topics in a way that everyone can understand, especially for audience who are not familiar with the topic. Meanwhile, students should not spend too much time explaining concepts that the majority of the audience already knows. | 3 |
| **SubTotal** |  | 15 |
| **Content/Coding** |  |  |
| System functionalities | Sufficient amount of coding effort and novelty should be reflected by presentations slides and oral presentation. Students are familiar with the subject topic of choice, and provides sufficient details to the topic. See suggested functionalities above for more details. | 15 |
| **SubTotal** |  | 15 |
|  |  |  |
| **Total** |  | **30** |

**Peer Evaluation Form**

Please submit this page **individually** to Blackboard. You don’t need to submit this if:

1. You did the project alone, OR
2. You are satisfied with the contributions of all the team members

1: No contribution

2: Little contribution

3: Moderate contribution

4. High contribution

5. Excellent contribution

**Evaluation (1~5)**

**Group Member \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_**

**Group Member \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_**

**Group Member \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_**