CS410 Project Proposal

1. **What are the names and NetIDs of all your team members? Who is the captain? The captain will have more administrative duties than team members.**
   Rui Liu (ruiliu7) Captain
   Zhenzhou Yang (zy29)

2. **What system have you chosen? Which subtopic(s) under the system?**
   We choose Option 2.2 Expert Search System.
   Both.
   We plan to finish an application where users can enter a URL link and our system will identify faculty directory pages, identify faculty webpage URLs on the directory website, and format a structured faculty list with relevant information from faculty bios if the given URL is valid.

3. **Briefly describe the datasets, algorithms or techniques you plan to use.**
   - The positive datasets are the faculty web pages given by students used for mp2.1 (https://docs.google.com/spreadsheets/d/198HqeztqhCHbCbcLeuOmoynnA3Z68cVxixU5vvMuUaM/edit#gid=0). The negative datasets would be non-directory web pages we collect online. Their ratio would be close to 1:1.
   - We will preprocess the html data and get their text. Using TF-IDF to filter the text and select features, vectorize the data into the input of our models.
   - We will train models using Python sklearn packages, such as linear regression, logistic regression, naive bayes, SVM, nearest neighbors and decision trees. Then we will choose the one yielding the highest accuracy as our final model.
   - If none of the models in sklearn perform well, we may build a neural network model including word embedding layer and linear layers.
   - If all the models failed to yield good accuracies, we may adopt several improving methods such as limiting the domains that users are intended to search.

4. **If you are adding a function, how will you demonstrate that it works as expected? If you are improving a function, how will you show your implementation actually works better?**
   We add a function to enable users to enter a URL link and do the crawling automatically. The way we demonstrate that it works as expected is that users will get direct feedback from our system if the entered URL is a directory page or not. If yes, the system will continue to search all URL embedded in this directory website and classify if any of them is a faculty URL page. If yes, we will add this website to the final structured output. Then users will be able to check the results from the automatic crawler.

   Besides, there are things we would like to improve. While we researched the provided system, we found out there are bugs that have not been catched. For example, if I enter some arbitrary meaningfulness word, the system returns back some information that is

not related at all. What's more, there are times that the faculty names are not displayed properly and listed empty. Therefore we would like to catch these bugs and fix them.

5. **How will your code communicate with or utilize the system? It is also fine to build your own systems, just please state your plan clearly.**
   We plan to maintain a similar user interface as the given system. However, our system will focus more on automating the crawling process of MP2.1. Users can enter a URL link and our system will identify faculty directory pages, identify faculty webpage URLs on the directory website, and format a structured faculty list with relevant information from faculty bios if the given URL is valid. In order to do the identification tasks, we plan to preprocess the text information on the input URL and do vectorization using TF-IDF, then train various models using sklearn packages and then select the best one as our model for the system. By doing this, our system will be able to classify any URL given by the users. To display the structured faculty information, we plan to utilize the format of the given system. Instead of using the interface directly, we will debug it first to take care of the blocks that should not be displayed when users enter meaningfulness words. What's more, we will fill in the information provided by our system to demonstrate our work.

6. **Which programming language do you plan to use?**
   - Machine Learning: Python scikit-learn package and pytorch
   - Web Crawling: Python selenium package
   - User Interface: Python in the given Github repo.
     - Supplement: Python tkinter, Javascript packages (haven't been decided)

7. **Please justify that the workload of your topic is at least 20*N hours, N being the total number of students in your team. You may list the main tasks to be completed, and the estimated time cost for each task.**
   Get familiar with the current system: 4hrs
   - Read related docs.
   - Clone repo and run the code.

   Get the dataset: 4hrs
   - Collect 900 non-directory dataset to make the ratio of positive and negative in directory dataset 1:1.
   - Collect 900 non-faculty dataset to make the ratio of positive and negative in faculty URL dataset 1:1.

   Preprocess data: 10hrs
   - Implement TF-IDF with various parameters to vectorize the data as the format to fit into the models and select the most optimal parameters.

   Develop models: 30 - 35hrs
   - Train models using Python sklearn packages, including linear regression, logistic regression, naive bayes, SVM, nearest neighbors and decision trees.
   - Build an neural network model using pytorch (haven't been decided)

- Fine tune the models and do cross validation and calculate accuracy for every model.
- Compare accuracy of different models and take the one with highest accuracy as the final model in our system.

Design and Implement User Interface: 5hrs

- Since the user interface of the current system was designed for words search, we need to customize the interface to better serve our needs.

Meetings: 10hrs

- Weekly meetings to keep both of the members on the same page.