CS-410 Text Information Systems: Final 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.

Name	**Net-Id**
Govindan Kutty Menon	gvmenon2
Harikrishna Bojja	hbojja2
Pushpit Saxena	pushpit2

Team Name: BayToBay
Team Captain: Pushpit Saxena (netid: pushpit2)

2. What system have you chosen? Which subtopic(s) under the system?

System	ExpertSearch System
Subtopics	1. Automatically crawling faculty webpages 2. Extracting relevant information from faculty bios 3. Stretch Goals: Topic Modeling on faculty bios.

3. Briefly describe the datasets, algorithms or techniques you plan to use

Techniques for Subtopic:

Automatically crawling faculty:

1) Users provide University URLs as input.
2) The websites (URLs) will be leveraged to crawl and collect the dataset for classifying “faculty directory pages” v/s “non-faculty”.
3) To collect the dataset for classifying "faculty webpage" v/s "non-faculty webpage", the faculty directory pages will be crawled.

Classification task:

1) Classify “faculty directory pages” v/s “non-faculty”
2) Classify "faculty webpage" v/s "non-faculty webpage"
3) We will try to build different classification models. Some of the models that we are planning to try and evaluate are **SVM, XGBoost, DL (hugging face transformers)** etc. We will leverage the URLs as well as text on the web-pages to extract features (vectorize text) to train the classification models.

Extracting relevant information from faculty bios:

1) Enhance Regular-expression to extract email-id from the bios
2) Enhance Named Entity Recognition (NER) to identify/ extract faculty name from bios.
3) Topic mining & keyword extraction on faculty bios information (stretch goal, if time permits).
4) We are planning to use Spacy, Gensim as well as Flair (BERT) for both NER models as well as topic modelling.

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?

Automatic Crawler:

1) We will provide an API/Console utility which can take an university home URL and then crawl the web-pages from that URL and return the list of faculty web-pages.

This will demonstrate that the crawler and classifier that we have built are working.

Web-pages classification tasks:

1) We will use some of the data from MP2.2 and MP2.3 assignment as suggested in the project topics document and demonstrate the performance of our models (F1 metric).

Extracting relevant information from bios pages:

1) As we are planning to use some recent and advanced NER models, we will show the difference in performance between regex based email/name extraction vs our NER model and also we will clearly state whether we are able to improve performance from a simple regex based approach or not.

Topic modelling:

1) We will demonstrate the top topics we can identify from the faculty bios dataset. Also, we will try to calculate topic coherence metric.

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

1) We will build our own system and will try to use the datasets provided by the existing ExpertSearch system (e.g. faculty bios, names, emails etc.) for training and evaluation.
2) We will also use the regex based NER model as a baseline to compare some of the more advanced NER models that we will try to train.

6. Which programming language do you plan to use?

Programming Language	Python, Javascript, HTML

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.

Number	Main tasks	Hours
1	Evaluation/ Analysis of the current system/ process and any baseline models already implemented in the system.	12 hrs.
1.1	Any required training data annotation	6 hrs.
2	Researching algorithms and overall system design	10 hrs.
3	Development of functionalities envisioned	
3.1	Recursive Crawler Implementation	10 hrs
3.2	Web Page Classification Model Implementation	10 hrs
3.3	Information Extraction	8 hrs
3.4	Topic Modeling	8 hrs
4	Self-evaluation and modification	24 hrs
5	Demo Preparation and Documentation	6 hrs.
6	Collaboration	3 hrs.
Total		96 hrs.