CS 410 Text Information Systems

Final Project Proposal by Team *Pinto Beans*

## ExpertSearch - Building a Physician Search Platform

### Overview

Patients are oftentimes faced with the challenge of finding a physician that is considered a good fit for the conditions they are experiencing. Consumers need to make multiple calls to different hospitals or practice groups and experience long wait times and scheduling difficulties. Today, no master physician roster is available for healthcare consumers, and hospital internal and external platforms lack interoperability.

This motivated the idea to build a platform to enable patients to easily find a provider that satisfies two criteria: 1) has a specialty that aligns with patient disease profile; 2) is within the patient’s geography, replacing a need for the patient to look through physician biographies within each individual hospital’s website. This platform, inspired by the ExpertSearch System, will provide the user with names, specialty, contact information, address of relevant physicians. It will further allow searching by hospitals and specific locations, for example, zip5 or county. The project falls under the free topic theme.

### Team Structure

| **Member Name** | **NetID** | **Role** |
| --- | --- | --- |
| Jia Lin Cheoh | jcheoh2 | Member |
| Seth Cattanach | sethac3 | Member |
| Akshat Agarwal | akshata4 | Member |
| Shengyuan Pan | pan43 | Captain |
| Arindam Saha | saha2 | Member |

### 

### Specification

The core methodology is to first build a crawler with seed pages to fetch physician biography pages from hospitals, practice groups, third-party websites, etc and parse/collect unstructured text data. An inverted index will be created by a back-end application to support efficient retrieval. Next, text similarity will be assessed using a vector space model and/or probabilistic language model to identify relevant physicians by specialty, conditions, procedures, locality, etc. Each physician description collected will be treated as a document, and standardized physician taxonomy the queries. A web application will allow users to interact with the system and perform queries.

### Datasets

* Health Care Provider Taxonomy Code Set version 22.0: This file consists of standardized physician taxonomy code, their classification, and description of their specialization. <https://www.nucc.org/index.php/code-sets-mainmenu-41/provider-taxonomy-mainmenu-40/csv-mainmenu-57>
* WebMd [https://doctor.webmd.com/results?q=&pagenumber=1&pt=41.3975,-90.8995&d=40&city=Illinois%20City&state=IL The seed keyword would be “illinois” for scraping.](https://doctor.webmd.com/results?q=&pagenumber=1&pt=41.3975,-90.8995&d=40&city=Illinois%20City&state=IL)
* Google business data <https://developers.google.com/my-business/content/offering-data>
* Physicians commercial data at Illinois (not open source) <https://doctordatabase.co/>
* CMS National Plan and Provider Enumeration System (NPPES): This file provides basic information about all organizations and individual providers with a National Provider Identifier (NPI). It consists of a list of registered healthcare providers identified via NPI number and their associated attributes including status, entity type, taxonomy, practice locations, etc.[[1]](#footnote-0) <https://download.cms.gov/nppes/NPI_Files.html>

### Implementation

Implementation is broken down into four components. Each one is separately explained below with estimated time and level of effort for justification. Python will be used for the backend application, and JS for the development of the GUI. The total amount of estimated time required sums up to 120 hours, exceeding the minimum required 100 hours. The resulting components of the system may be hosted on a public provider, such as AWS or GCP, to enable efficient development and integration of the different application components.

1. **Web scraper**

Description: This is the implementation of the text retrieval/web search component of the core methodology. We will build a crawler to web scrape physician information within the larger Chicago or IL geography. The goal is to collect data on physician specialty, contact information, practice group location, etc

Input sources (list of “seed” URLs, not limited to):

* + 1. Physician biography pages in hospitals’ websites
    2. WebMD using seed keyword Chicago or Illinois.
    3. Physician business details on Google Business
    4. (Optional) Physicians commercial data at Illinois

Output: unstructured text data

Estimated time required: **30 hours [= 15(scraper) + 15(cleaning)]**

1. **Inverted Index (back-end)**

Description: this component creates an inverted index from the web scraper data to facilitate efficient text retrieval by subsequent components of the system

Input: web scraped text data (documents/web pages) from the web scraper (described above)

Output: Inverted index that stores text data in an efficient manner for subsequent retrieval

Estimated time required: **20 hours**

1. **Text similarity/relevance analysis (back-end)**

Description: Adopt a VSM or probabilistic language model to analyze retrieved data and find the relevant physicians. We can also potentially adopt a cluster approach to convert the text information into features and utilize classification models to group physicians by specialties, patient conditions, locality, etc

Input: scraped unstructured text

Output: A list of relevant physicians with corresponding attributes from scraped data and enriched by NPPES

Method of Evaluations - Evaluate via precision-recall techniques. Also use of relevance feedback and pseudo feedback.

Estimated time required: **30 hours**

1. **Front-end to present and host the final product**

Description: A minimalistic web application that serves as an interface for end users to search plain text documents. It will interact with the backend that leverages the inverted index in order to present results that best match the user’s keywords. This will be built with ReactJS.   
Input: Keywords from the user

Output: Matching documents

Estimated time required: **25 hours**

1. **Other Miscellaneous**

Create gold standard dataset for evaluation: **10 hours**

Write up final documentation and come up with a presentation: **15 hours**

### Evaluation

We will manually create a labeled dataset including physicians and their “gold standard” specialties. This will be done by members reading through the website descriptions and making judgment calls. This dataset will be used to evaluate the accuracy of the recommended list of physicians deemed most relevant by the models of choice.

1. The NPPES file could be used as a layer of enrichment in the final version of the retrieved, filtered, and cleaned physician information. For example, the file contains both primary and secondary practice locations, so it can be used to identify alternative locations at which the identified physician practices. This feature can enable the patient to choose among a list of potential locations for the visit. [↑](#footnote-ref-0)