

Documentation for CS410 Final Project: Improving ExpertSearch System

Mengyu Zhai (NetID: mzhai)

Fall 2020

1. Overview

1.1 Background

This project is for Fall 2020 CS410 Text Information Systems final project option 2.2 ExpertSearch System. The information quoted below is from CS410 course project instructions and what contained is the foundation of the current project:

"The ExpertSearch system (<http://timan102.cs.illinois.edu/expertsearch//>) was developed by some previous CS410 students as part of their course project! The system aims to find faculty specializing in the given research areas. The underlying data and ranker currently come from the MP2 submissions of the previous course offering. You can read more about it [here](#) (Sections 3.6 and 4: Project are especially relevant). The code is available [here](#)."

1.2 Functions of Code

The current project is trying to enhance the utility of the ExpertSearch System by extracting relevant information from faculty bios. More specifically, the code uses techniques for extracting other information, i.e., faculty research interests, than what is already provided in the original system.

Simply put, topic mining is performed on bios and top-keywords are found and shown as the common research areas, in addition to the bio information shown in the search result.

This added function does not influence the ways that users can use to conduct searches in the system. Users can search the expert by whatever search key word they want to use and applying whatever location or university filters they choose. Now if the user is especially interested to know which experts work in a certain research area and what their respective research interests are, they can directly search by that research area they have in mind. What is improved is that in the search result, rather than just showing the matching parts in the bio of an expert (or matching parts in the bios of all the experts in the search results) as before, a set of highly possible research interests (using top-keywords) of the expert will also be

provided. The method is not perfect as we assume top-keywords are common research areas but cannot guarantee that is always the case.

2. Implementation

2.1 What Are Used

- Python
- LDA topic model: generative statistical model – to detect topics
- [gensim API](#): to lemmatize and add bi-gram tokenization (build dictionary), and train LDA model
- [pyLDavis](#): to provide interactive visualization of LDA topic results
- [word_cloud](#): to visualize corpus key words
- Other major tools/ packages involved: Jupyter notebook, numpy, pandas

2.2 Steps

- With jupyter notebook, utilizing gensim API and LDA to build the topics from the combined bios (10 for each bio).
 - Preprocess data
 - Lemmatize and stop word removal
 - Build dictionary and use word_cloud to visualize the corpus
 - LDA modeling
 - Visualize LDA topics with pyLDavis
- Then also within jupyter notebook, results are saved into a researchinterest file under ExpertSearch\data.
- Finally modify the original ExpertSearch web application source code to add and display top topics as research interests.

2.3 Results

The corpus outlook built using [word_cloud](#):

d 0

Topic 5 (weight 0.58): "science" + "university" + "research" + "professor" + "student" + "award" + "department" + "cs" + "fall" + "phd"

Topic 4 (weight 0.24): "software" + "programming" + "work" + "research" + "language" + "design" + "security" + "program" + "project" + "application"

Topic 7 (weight 0.17): "ieee" + "network" + "pp" + "international" + "conference" + "communication" + "vol" + "wireless" + "design" + "symposium"

Topic 8 (weight 0.01): "engineering" + "research" + "university" + "science" + "professor" + "electrical" + "award" + "student" + "faculty" + "department"

Processed Document

tarek home page tarek professor willett faculty scholar department science university illinois urbana champaign urbana il tel fax receive ph university michigan arbor professor kang shin asistant professor university vi august join university illinois urbana champaign associate professor tenure professor lie primarily include operate networking sensor network distribute embed real time especially interested develop theory architectur compute abstraction predictability software motivate increase software complexity grow source determinism application range sensor network large scale server farm transportation medicine new navigation green gpt research cyber physical compute group paper professional activity teach example cyber physical operate web architecture advance operate network deeply embed network mail letter cs dot uiuc dot thank visit webps department_science university_illinois urbana_champaign urbana_il tel_fax receive_ph university_michigan university_illinois urbana_champaign associate_professor sensor_network real_time especially_interested architectural_support sensor_network large_scale cyber_physical professional_activity cyber_physical cs_dot

Another example of some selected topics for a different bio:

d 760

Topic 4 (weight 0.38): "learn" + "model" + "pdf" + "machine" + "machine_learn" + "problem" + "language" + "data" + "datum" + "base"

Topic 8 (weight 0.21): "research" + "university" + "science" + "engineering" + "professor" + "award" + "department" + "student" + "cs" + "publication"

Topic 1 (weight 0.19): "software" + "engineering" + "ieee" + "software_engineering" + "communication" + "radar" + "ku" + "nav" + "pp" + "signal"

Topic 2 (weight 0.18): "design" + "conference" + "international" + "proceedings" + "performance" + "architecture" + "memory" + "symposium" + "ieee" + "c"

Topic 9 (weight 0.01): "research" + "engineering" + "student" + "science" + "program" + "faculty" + "graduate" + "university" + "course" + "information"

Topic 0 (weight 0.01): "pdf" + "paper" + "graphic" + "acm" + "security" + "siggraph" + "video" + "light" + "visualization" + "symposium"

Topic 6 (weight 0.01): "ieee" + "conference" + "international" + "network" + "pp" + "acm" + "wang" + "data" + "proc" + "transaction"

Processed Document

prof babu department science engineering indian institute technology hyderabad mail lith ac education bombay mtech university madras diploma vvsr polytechnic research big data analytic graph theory algorithms h distance run mountaineering publication satya trinadh seetal potluri shankar balachandran ch babu kamakoti xstat statistical algorithm peak capture power reduction scan test accept satya trinadh seetal potluri ch bal efficient heuristic peak capture power minimization scan base test low power electronic ravindra guravannavar ajit diwan ch sort order interesting vldb bharat adsul ch babu jugal garg ruta mehta milind sohoni simplex market sagt bharat adsul ch babu jugal garg ruta mehta milind sohoni nash equilibria fisher market sagt ch babu ajit diwan degree condition forest graph discrete mathematics ch babu ajit diwan oriented forest direct electronic note discrete mathematics ch babu ajit diwan subdivision graph generalization path cycle discrete mathematics special issue honor miklos simonovits birthday ch babu ajit diwan disjoint cycle chord graph ac graph theory ch babu ajit diwan subdivision unicyclic graph accepted graph combinatoric ch babu degree condition subgraph ph thesis teach linear optimization predictive analytic business analytic algorithm visi cad data structure algorithm update feb department_science engineering_indian institute technology big_data graph_theory low_power discrete_mathematics discrete_mathematics discrete_mathematics special_issue g ph_thesis predictive_analytic business_analytic discrete_structure data_structure

The top topics with the highest weight get saved to researchinterest.txt and will be the one to display in expert search results.

Search by key words, you will be able to see the results similar to image below:

NLP

Dragomir R. Radev

Computer Science, Yale University

Connecticut, United States

Topic 4 (weight 0.87): "software" + "programming" + "work" + "research" + "language" + "design" + "security" + "program" + "project" + "application"

...html bib pdf), graph-based **nlp** and ir (html...bib pdf), **nlp** for bioinformatics (html...pdf), deep learning for **nlp** (html bib pdf

Langlais

Computer Science, University of Montreal

Québec, Canada

Topic 6 (weight 0.45): "conference" + "proceedings" + "international" + "human" + "journal" + "workshop" + "paper" + "language" + "interaction" + "acm"

langlais, philippe natural language processing (**nlp**) is a field of...human (natural) languages. challenges in **nlp** are numerous and enrol...as well as to develop **nlp** applications. the latest applications

You can also filter by locations and/or universities:

NLP

Locations

× Singapore

Universities

e.g. Stanford University

Apply Filters

Sun Aixin

Computer Science & Engineering, Nanyang Technological University (NTU)

Singapore, Singapore

Topic 5 (weight 0.51): "science" + "university" + "research" + "professor" + "student" + "award" + "department" + "cs" + "fall" + "phd"

...xiaoyan zhu. www. apr 2019 **nlp** learning travel time distributions...shafiq joty. ijcai. july 2018 **nlp** neuirec: on nonlinear transformation...chenliang li. dec 2018 ner **nlp** ir selected publications news

Erik Cambria

Computer Science & Engineering, Nanyang Technological University (NTU)

Singapore, Singapore

Topic 5 (weight 0.48): "science" + "university" + "research" + "professor" + "student" + "award" + "department" + "cs" + "fall" + "phd"

...and researches about ai and **nlp**. prior to joining ntu,

5

Now compare with the original system's results below, which doesn't provide top topic key-words:

NLP

Dragomir R. Radev

Computer Science, Yale University

Connecticut, United States

...html bib pdf), graph-based **nlp** and ir (html...bib pdf), **nlp** for bioinformatics (html...pdf), deep learning for **nlp** (html bib pdf

Langlais

Computer Science, University of Montreal

Québec, Canada

langlais, philippe natural language processing (**nlp**) is a field of...human (natural) languages. challenges in **nlp** are numerous and enrol...as well as to develop **nlp** applications. the latest applications

3. Installation and Run

Codes uploaded to github include ExpertSearch_LDA.ipynb and web application codes in ExpertSearch.zip.

Be aware that the web application codes does not work on Windows (mainly because gunicorn is not supported on Windows)!

3.1 Simple Instructions (If you are familiar with the project)

To run the web application code, run the following command from ExpertSearch (work with Python2.7 on MacOS and Linux):

```
gunicorn server:app -b 127.0.0.1:8095
```

The site should be available at <http://localhost:8095/>

3.2 More Detailed Instructions (If you are not familiar with the original project)

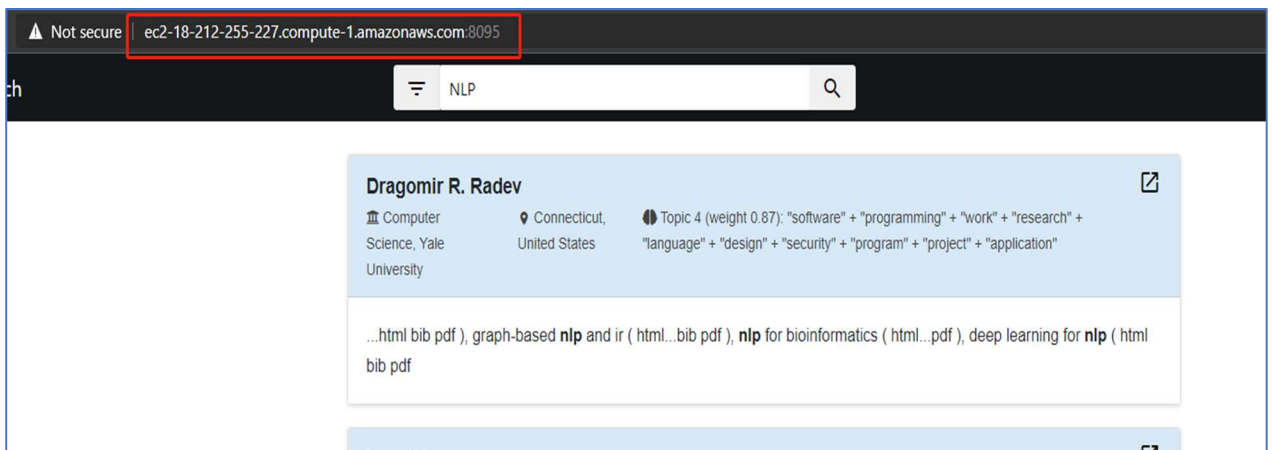
- Download ExpertSearch.zip.
- Unzip it in Python2.7 on MacOS and Linux.
- Go to the folder and run the following command:

```
[ExpertSearch]$ gunicorn server:app -b 127.0.0.1:8095
```

- You should see something similar to below once the website is running:

```
[ExpertSearch]$ [2020-12-11 03:07:57 +0000] [17325] [INFO] Starting gunicorn 19.10.0
[2020-12-11 03:07:57 +0000] [17325] [INFO] Listening at: http://127.0.0.1:8095 (17325)
[2020-12-11 03:07:57 +0000] [17325] [INFO] Using worker: sync
[2020-12-11 03:07:57 +0000] [17329] [INFO] Booting worker with pid: 17329
```

- The site should be available at <http://localhost:8095/>. Otherwise, you can open the port 8095 so people can access it from <http://yourdomain:8095>, for example:



References:

1. [Reference code]: <https://github.com/CS410Fall2020/ExpertSearch/>
2. [Reference code]: <https://github.com/TeddyWang0202/BeyondLD>
3. [Reference file]: <https://bhaavya.github.io/files/SIGCSE2020.pdf>