CS 410 Project Documentation

Overview

Our project involved extending a Text Mining System created by Bhaavya (https://bhaavya.github.io/), by implementing several new features to extend its functionality. The existing system already allowed users to generate query and search using that query in Kibana interface for related keywords based on a corpus of historical newspaper articles. The purpose of this system was to aid in social science research by finding relevant strings of words in order to quickly find historical information.

We extended the system by adding “Topic Mining" feature. This new feature provides user with the option to upload multiple .txt files to generate topics using an LDA algorithm and an interactive visualization for the number of topics provided as input by the user. Clicking the submit button takes the user to the visualization page which displays the output generated by the LDA in a clear and concise manner. The user is then able to download the visualization. Users can browse through the different topics, adjust model parameters, and explore the visualization to find relevant information.

Scope Change due to production system unavailability (https://timan102.cs.illinois.edu/)

Initially we planned to do topic modelling on the historical newspaper data which this project had but due to unavailability of the system we had to change the scope after discussion with Bhaavya. We had the discussion with Bhaavya and kept the professor in the loop.

Changed scope:

Allow user to upload their own documentation and generate visualization on top of that instead of using the existing data. We also had periodic reviews with Bhaavya and made sure incorporate her suggestions. The implemented feature is modular enough so that it can easily extended to achieve the topic modeling for exiting documents in system’s database.

Implementation Documentation

We spend a major portion of our allocated time understanding the existing code base which was required for configuring and enhancing the system with new functionality. As the server on which the Text Mining System was initially hosted is temporarily offline due to maintenance, we had to explore the option to set up the system locally which was not part of our initial plan. The majority of code changes were done to the following files to add topic modeling functionality.

- base.html contains the main page which links to Topic Mining page
- topic_mining.html contains the Topic Mining web page
- topic_modeler.py generates and returns the visualization to display as an html
- word2vec.py is runs the entire application

Usage Documentation

There are several ways to run the code depending on user preferences. Generally, we recommend creating a python virtual environment, installing pip and the relevant packages necessary to run word2vec.py using some sort of terminal or command line. If there is anything missing that is not covered by this list, simply install whatever package is missing that throws an error.

- pip install --upgrade pip
- pip install virtualenv
- pip install flask
- pip install gensim
- pip install flask_session
- pip install pyLDAvis
- pip install nltk
- pip install pandas
- pip install metapy pytoml
- pip install matplotlib

When you are done, simply navigate to judel/tms and run python word2vec.py.

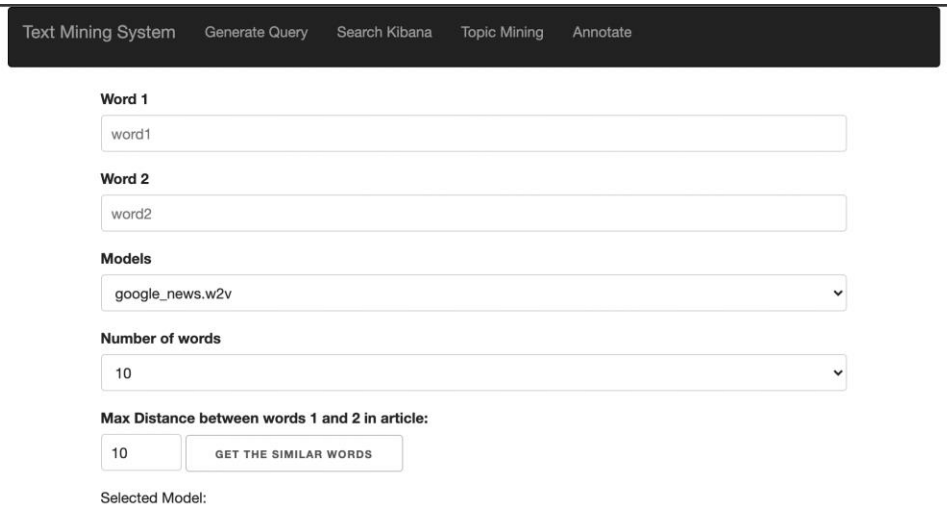
Commands used:

→ % cd <folder path>/ judel/tms

→ tms % python word2vec.py

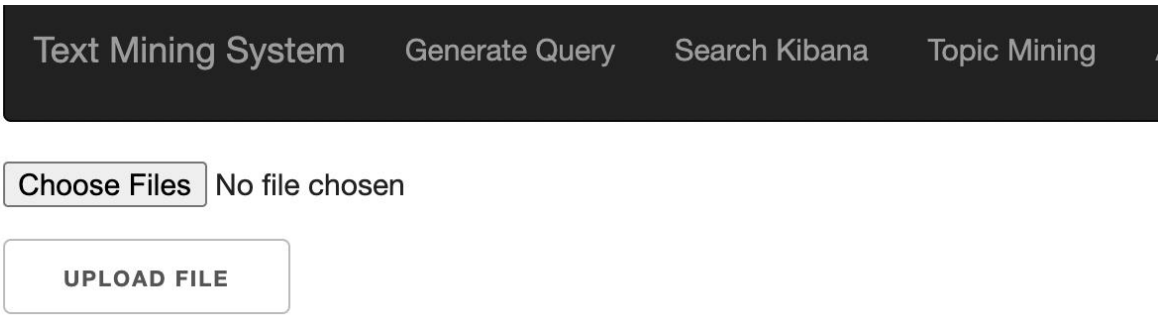
After successfully running it, navigate to http://localhost:6600.

It may take some time to load the webpage, but when it does navigate to the **Topic Mining tab**.



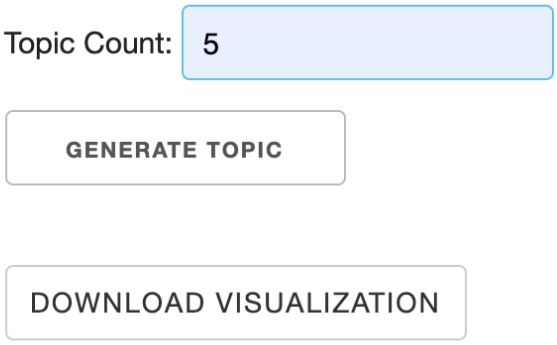
Upload functionality

This tab contains an upload button that allows the user to select several .txt files to upload for generating topics.

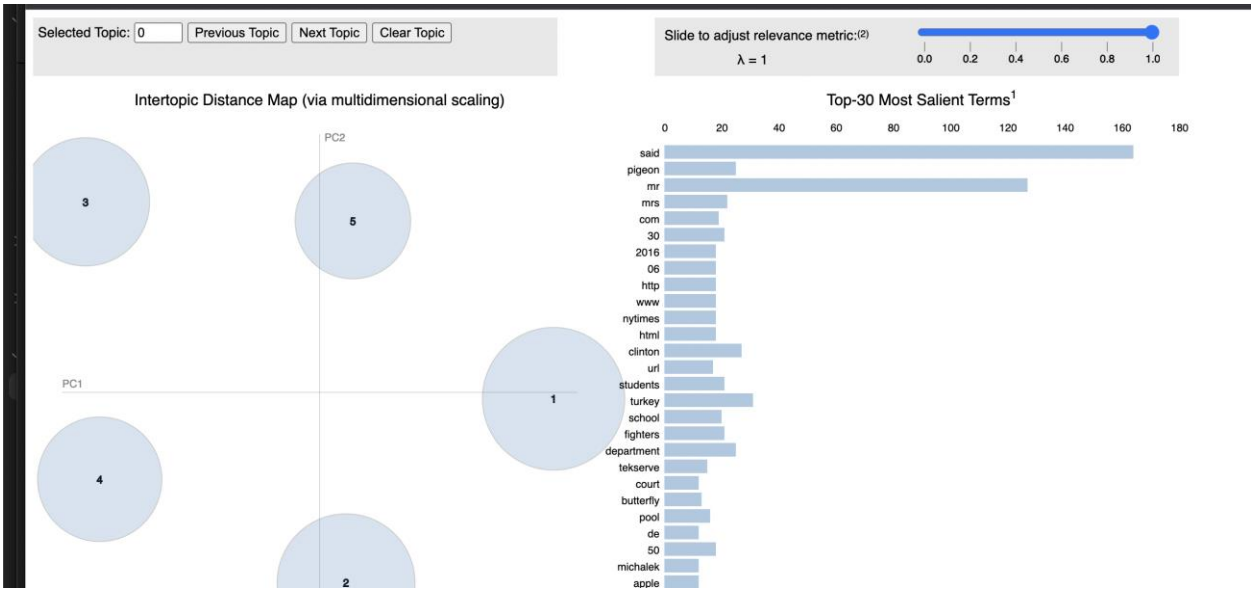


Generate Topic & Visualization

Clicking the “generate topic” button returns a visualization which can be downloaded.



The visualization has buttons to cycle through the topics, clear the topic and adjust the relevance metric. The user is also able to interact with the distance map and bar graph to obtain further information.



Contribution

Nivedita Chatterjee (nc19) - visualization development, back-end functionality, documentation
Sudhendu Sharma (sharma78) - visualization development, back end & front-end functionality
Yuriy Kotskyy (ykotsk2) - front end functionality, documentation, project proposal
Captain: Yung Chieh Huang (ych10) - front end functionality, presentation, progress report

Future scope

- Topic Coverage for the uploaded document
- Integration with Production system (Could not be done due to Production is down for maintenance currently)
- Time series analysis (Could not be done due to Production is down for maintenance currently)
- Multiple Model Choice rather hardcoded LDA
- Multiple user support concurrently