



Monitoring and mitigating misinformation that disrupts adaptation to climate driven health threats

Students: Roberto Lopez-Trigo, Colleene McCurdy, Nathan Chiche, Brian Rodriguez, Kyle Ghani, Kyle Zimmer, Javier Oliva, Jeffrey Quispe, Khizar Nasir, Gabriel Prieto

Mentor: Sam Malloy, Mowafak Allaham, Ayse Lokmanoglu

Instructor/Faculty: Masoud Sadjadi, Florida International University

PROBLEM

Misinformation has a great impact on the public since people are obtaining their information mostly from social media and other online sources. Navigating this chaotic space is overwhelming. People unknowingly spread information from these questionable sources, magnifying the problem. One of the most concerning areas of misinformation is climate change.

In order to shed light on the issue, our team has compiled engagement data related to climate news articles from Newswhip. Using this data we are able to analyze trends and point out topics of concern.

CONTRIBUTION

Database:

- Created an Entity Relationship Diagram to reduce redundancy and separate the data into logical entities with relationships
- Conducted exploratory data analysis to detect inconsistencies such as having columns that should be unique but were not, null values and repeated tuples.
- Created a python script that generates a PostgreSQL database from a csv file using multiple libraries such as psycopg2, SQLAlchemy, pandas, etc.

Machine Learning:

- Implemented a Jupyter notebook to do exploratory data analysis of the NewsWhip data to figure out what kind of clustering method is appropriate, thus we found out that we should work with the headline and the body of the article to try to cluster them into meaningful groups.
- Created another Jupyter notebook that implemented the text preprocessing steps and researched different ways to accomplish word embedding
- Implemented in a Jupyter notebook a proof of concept of topic modeling using BERTopic for the first 2000 documents
- Worked on a claim algorithm that uses a pretrained PyTorch neural network to determine what kind of claims each article is making.

Feature:

- Worked on creating a 3d bubble chart to compare all types of engagement
- Created a version of the app that uses the created database instead of working with csv files, thus requiring to write SQL queries tested in PGAdmin .
- Worked on changing the Information Trends page's UI to work with BERTopic so that the user can find articles according to search term and see similarities in the data
- Worked on putting article cards color coded by article engagement with a button that redirect the user to the article page

CURRENT SYSTEM

The monitoring and mitigating information project aims to spread awareness about climate misinformation and the public perception of climate related issues.

User's can focus on multiple sources and filter the data according to their needs. This data could be filtered by engagement type, publication date, and top articles. That data can then be analyzed using pie charts, histograms, bubble charts and line graphs. We used BERTopic to cluster the data into meaningful groups allowing us to find topics related to a search term and thus the articles related to that topic.

Various climate disaster types can also be found on a United States map with the number of incidents separated by states. Estimates of the public's opinion on certain climate questions can also be seen on the survey data page. A breakdown of public interest into climate news can be found on the climate relevancy page.

REQUIREMENTS

Hardware:

- Mac
- Linux

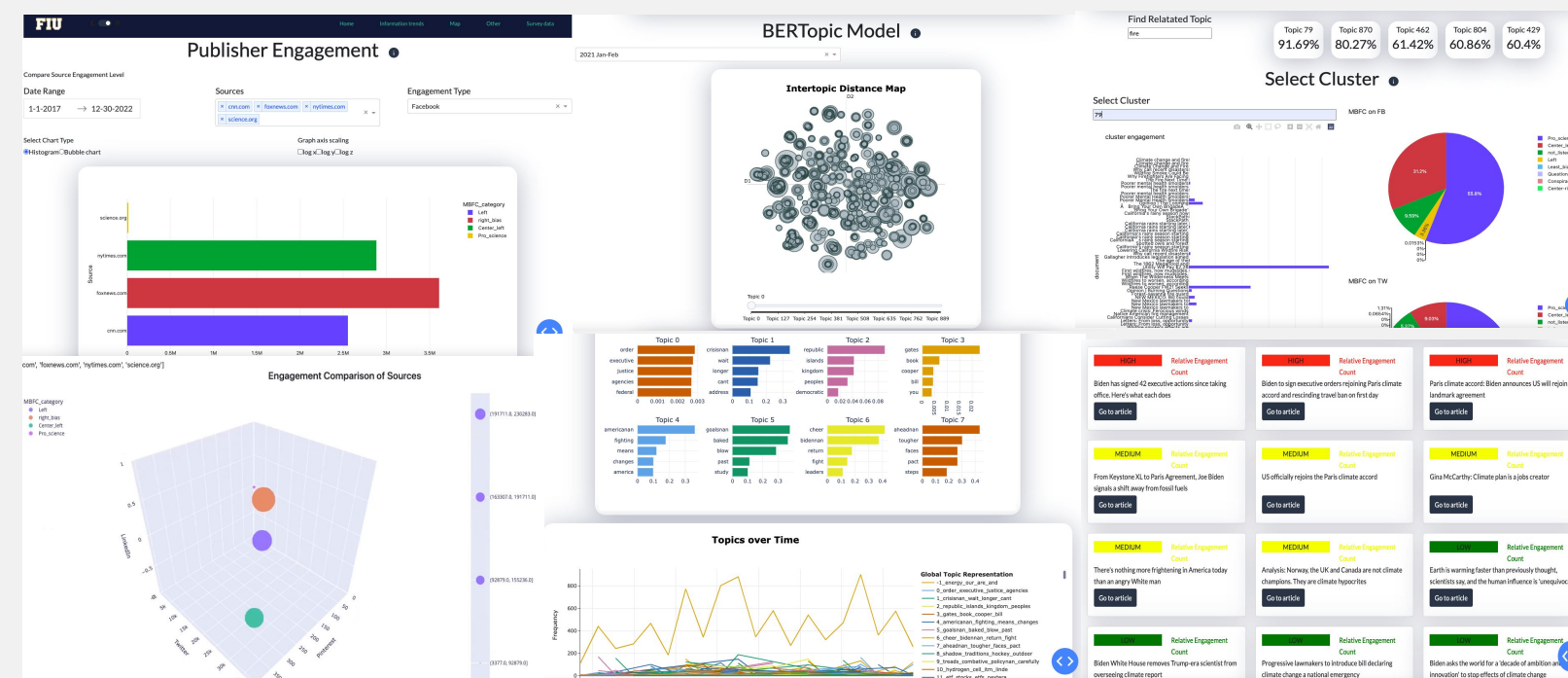
Software:

- Python
- PostgreSQL
- BERTopic
- DASH

Version Control:

- GitHub (<http://www.github.com>).

DASHBOARD



SUMMARY

We compiled the work from the previous capstone iteration and turned it into a DASH multi page application with analytical functions and visual aids in the form of dynamic graphs. We used BERTopic to cluster the articles in meaningful groups to find relevant documents to the user's searched term. All the Newswhip data static files that were being used as a reference for the scripts were put into a Postgres database. We tested the database by building a proof of concept using optimized queries.

We created a multithreaded processing scraper using beautiful soup. All UI features were added during this terms iteration. BERTopic used to cluster articles based on their text and headlines. Github pages were deployed and used for collaboration. Database was created, connected to and queried to from application. Further iterations will make an application that is more depended on the database.

IMPLEMENTATION



Google Docs - Used to keep track of everyone's contributions to the project. Multiple meetings and their information has been recorded through Google Docs.



Github - A site we used to access our code. For the group to test and add changes to the project.



Plotly - Allows a point & click interface. Assists in creation of our dashboard.



PostgreSQL - Relational database management system.



Python - The main language that we used for this project.



BERTopic- Topic modeling technique we used to cluster articles based on their text and headlines.

REFERENCES



ACKNOWLEDGEMENT

The material presented in this poster is based upon the work supported by Sam Malloy. We thank Sam Malloy, Mowafak Allaham, and Ayse Lokmanoglu for their mentorship and assistance that we received throughout the senior design project.