



**Knight Foundation
School of Computing
and Information Sciences**

Knight Foundation School of Computing and Information Sciences

Summer 2022 Senior Design Project

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

SQL Injection Attack- Being a cyber security major, I informed the team on application vulnerabilities such as SQL injections. I used a tool called sqlmap to detect possible sql injections. After finding possible vulnerabilities, query inputs were sanitized by writing parameterized statements.

```
sql=f'''
SELECT chosen.source_publisher,
chosen.mbfc_category,
CASE
    WHEN '{engagement_selected.replace('.', '_')}='fb_data_total_engagement_count'
    THEN SUM(chosen.fb_data_total_engagement_count)
    ELSE SUM(chosen.tw_data_tw_count)
END

```



```
# determine entities attributes
facebook = ['uid', 'fb_data.total_engagement_count', 'recent_fb_counts',
            'fb_data.shares', 'fb_data.total_count_delta', 'fb_data.delta_period',
            'fb_data.delta_period_unit']

twitter = ['uid', 'recent_tw_counts', 'tw_data.tw_count',
            'tw_data.total_count_delta', 'tw_data.delta_period',
            'tw_data.delta_period_unit']

linkedin = ['uid', 'li_data.li_count',
            'li_data.total_count_delta', 'li_data.delta_period',
            'li_data.delta_period_unit']

pinterest = ['uid', 'pi_data.pi_count', 'pi_data.delta_period',
            'pi_data.delta_period_unit']

source = ['source.publisher', 'source.domain',
            'source.link', 'source.country', 'source.country_code',
            'source.language', 'MBFC_category', 'url_to_test']

```

Database- I assisted the database team in indexing the database by creating a uuid. I also helped identifying a proper primary key, foreign key and attributes for the different entities. For database security I discussed how to implement a least privilege approach when assigning users to the database when it is put up on FIU's VM.

```
def CreateAggregates(filename):
    #Read Csv and index by date_time
    df = pd.read_csv(filename, encoding='Latin1')
    df = df[NEEDED_COLUMNS].copy()
    df.index = pd.to_datetime(df['date_time'])

    #group data by month and sum up articles that occurred each month using .count()
    countDF = df.groupby(pd.Grouper(freq='M')).count()
    countDF = countDF[['date_time']].copy()
    countDF.rename(columns={'date_time': 'articlecount'}, inplace=True)

    #group data by month and sum up engagement on a separate DataFrame
    df = df.groupby(pd.Grouper(freq='M')).sum()
    #Merge the two data frames
    df = df.join(countDF)

    return df

```

Aggregate Data- I assisted our team leader in creating a script to sum up the total number of engagement with climate articles per month from our sources for 5 years worth of data. We then compared climate engagement to total source engagement to end up with a percentage of climate engagement to overall engagement.

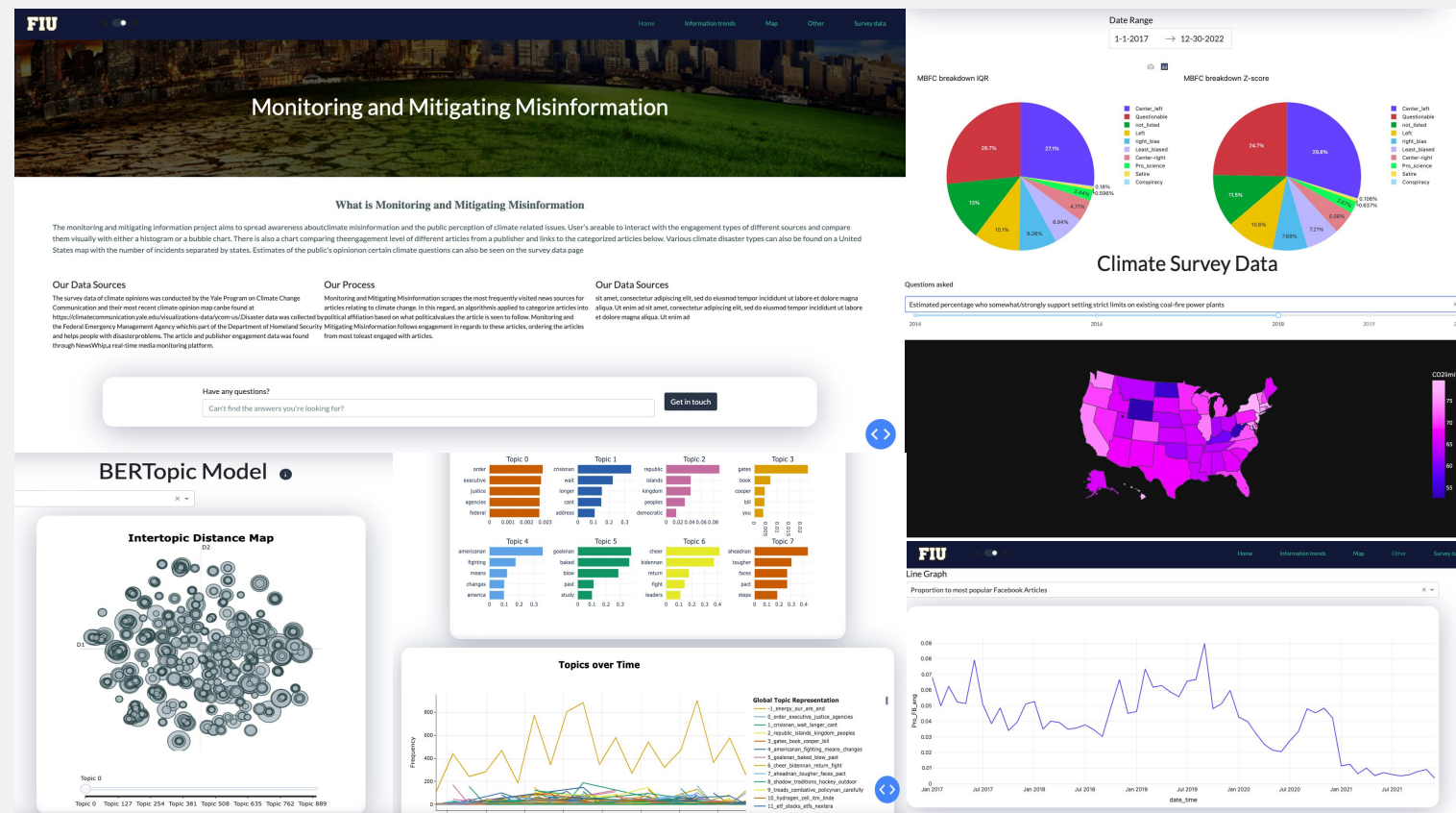
CURRENT SYSTEM

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.

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.

REQUIREMENTS

Hardware:

- Mac
- Linux

Software:

- Python
- PostgreSQL
- BERTopic
- DASH

Version Control:

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

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.



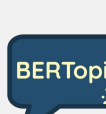
Plotly - Allows a point & click interface. Assists in creation of our dashboard.



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



**Engineering
& Computing**

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.

FLORIDA INTERNATIONAL UNIVERSITY