***Introduction***

There are many moving parts that have been succinctly put together for a successful final product, *Sentiment Analysis.* The purpose of the application is to analyze news articles in specific categories from various sources to convey the overarching sentiment of the publishers. In the application, the dashboard serves the purpose of displaying all the information that has been gathered through the three main parts of the application. The first part of the application has been coded to efficiently extract news articles, through the use of APIs, in the defined categories and remove articles not pertaining to the news categories of interest. The second part of the application is linked to cleaning and storing of data in the SQL database. In combination with the first part of the application, the application cleans the data and extracts the information relating to the overall sentiment of the article and specific keywords that were analyzed in the generation of the sentiment. The last part of the application is related to the extraction of the data from the database into the dashboard for its presentation in a manner that allows us to draw conclusions regarding the sentiment and keywords of a particular topic.

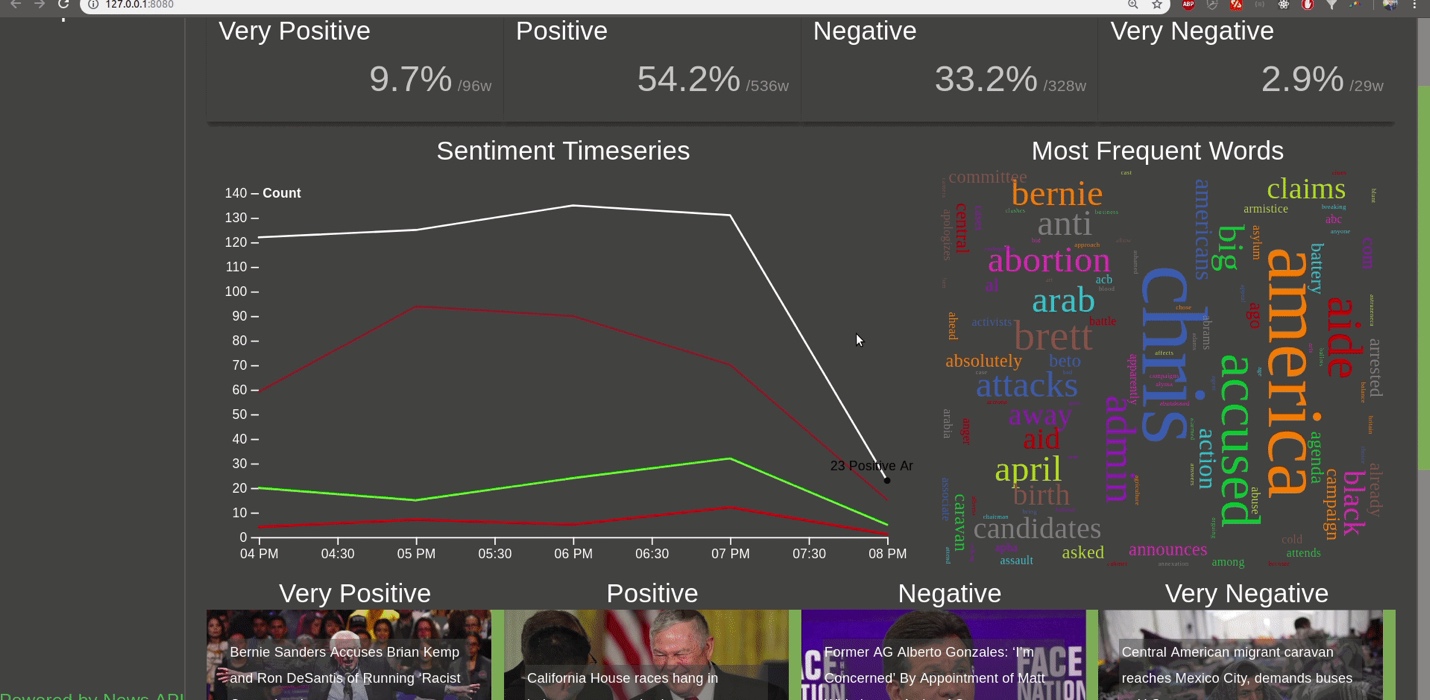
***Data Source & Modules***

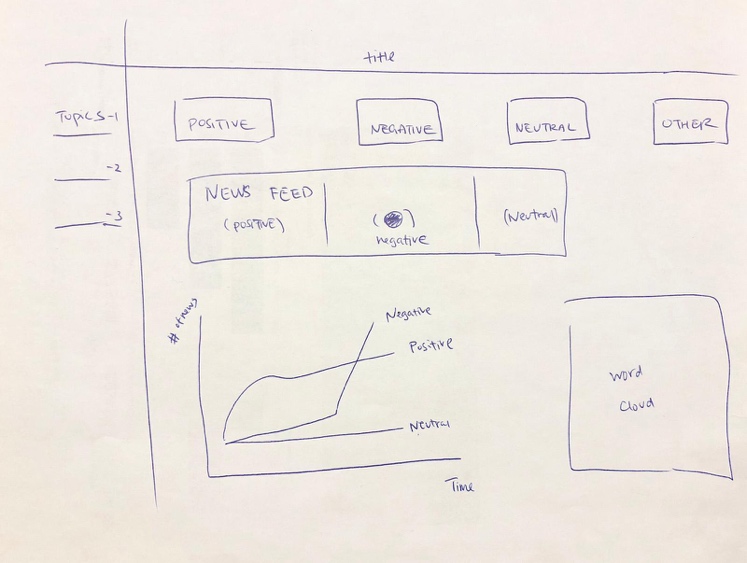
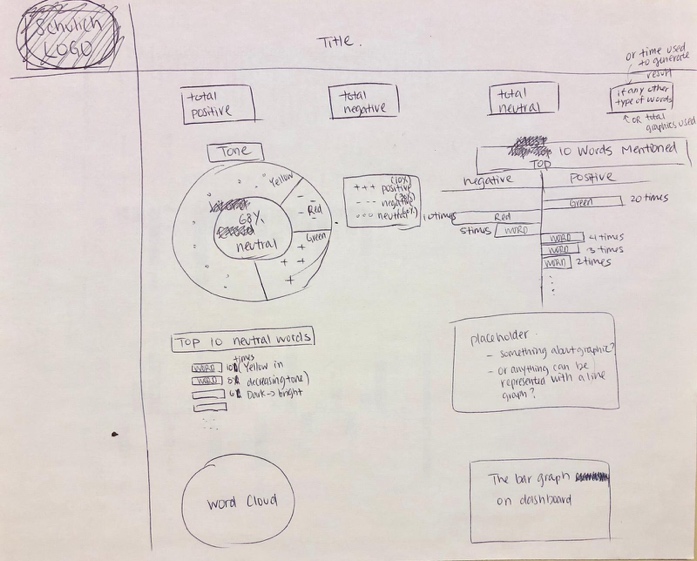
The main sources of data include news websites that are directly linked to the application with the help of APIs to extract and analyze news articles for the dashboard.

We have used many built-in modules in Python to successfully develop and run the application, but main ones include Text Blob, NLTK, and Flask. These modules were crucial in the development of the application and their main functionality within the application were as follows:

1. Text Blob: The main purpose of Text Blob is to process textual data. It provides the users the API to effectively utilize Natural Language Processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis among others. Our team utilized Text Blob for the purpose of understanding the sentiment of each article and then to create an overall sentiment of the news category. In the sentiment analysis library of the module, the sentiment property returns a named tuple with two variables, polarity and subjectivity. We used the polarity score to understand the sentiment of the articles and used SQL as data storage.
2. Natural Language Toolkit (NLTK): It is a leading platform for building Python programs to work with human language data. NLTK supports classification, tokenization, stemming, tagging, parsing, and semantic reasoning functionalities to allow the users to process text and derive insights from it. In the process of developing the Sentiment Analysis software, our team utilized the Stop Word Library within the module to clean verbiage and/or words that are not helpful in deciphering the sentiment of the text. It was intertwined with the Text Blob Module.
3. Flask: It is a framework that is used to develop web applications. The framework was used develop the web interface and a medium between the SQL database, Python Code and web application. It was used in rest\_api.py to connect the data from SQL to the web application in an organized manner.

***Mock-Up Vs. Dashboard***



******

Final Version

Mock-Up One Mock-Up Two

There are many differences in the layout of the dashboard from when we initially started the development of the application. In the initial design, we had four (4) different variables and information points in the dashboard with different type of information. As we progressed in the development of the application, it became evident that some of the things that we were trying to accomplish are difficult and would require a higher level of expertise. The initial design had the following visuals:

1. Table Chart: It would provide the sentiment analysis of the news category, Positive, Negative, Neutral, or Other.
2. Pie Chart: It would show the proportionate divide of the sentiment for each category to visual analysis and easier understanding of the data.
3. Most Frequent: It would show the cloud of words that are utilized in that particular news category.
4. Scroll Bar: A news scroll bar to show all the different news that are being used in creating the sentiment.

In the current Dashboard, there are four main features of the dashboard that have been integrated into the application for the analysis of news sentiment and words frequency. Each part of the dashboard serves a different purpose and allows the users to draw various types of conclusions. It includes:

1. *Table Chart*: The purpose of this part of the dashboard is to allow users to draw conclusions about the sentiment of the topic through four categories, Very Positive, Positive, Negative, and Very Negative. It allows the users to understand the overall tone of the topic in a segmented manner, allowing them to shape their understanding of the topic accordingly. The data for *Table Chart* is driven from the articles that are extracted through the use of APIs and a corresponding list of words that are categorized as very positive, positive, negative, or very negative for sentiment analysis.
2. *Sentiment Time Series*: The second part of the dashboard, *Sentiment Time Series*, allows the users to visually understand the changes in the sentiment of the news articles over time in a particular category. Furthermore, it can also allow the users, for example, the Journalists, to utilize the given information in articulating their thoughts and publishing their articles to effectively convey their sentiment in the article. It utilizes the data, gathered from the articles, to publish the varying sentiments of a news category.
3. *Most Frequent Words*: Similar to *Top 10 Words based on Sentiment,* it will allow the users of the dashboard to understand the keywords that are being utilized in attracting the readers towards the topic of interest. For example, it can help the users, journalists, of the dashboard in optimizing the use of words in their own article to most effectively convey the message as well as to drive traffic towards it.
4. *News Feed*: It showcases all the news articles that are being utilized in the analysis and presentation of the data. It will allow the users of the dashboard to analyze the articles for any underlying biases that are inherently present in the dashboard and/or read the articles to grasp a better understanding of the context. Additionally, the articles are categorized into the four unique sentiment categories for additional context.

***Proposal on Model Training***

There are four key indicators in the dashboard that our team has created, and we think the best indicator to train for future predictions would be the *Sentiment* *Time Series.* It will allow the users to understand the changing behavior of the public about a news topic and allow the readers to make informed decisions based on data and actual sentiment analysis in comparison to just “gut feeling.” We would utilize the Sklearn Module to train the model on the four types of sentiment in different news categories and create a trail of words, sentences, and patterns for the model to recognize the varying types of sentiments in the news articles. Furthermore, we would utilize the model to learn the changes in sentiment during different times of the day and inform its users on the predicted changes throughout the day.