William Hiatt
Deven Biehler
Gabe Matthew
Sprint 2 Report
Sprint Dates: February 20th - March 24th

Sprint 1 Retrospection: The majority of the tasks during sprint 1 involved developing a conceptual model, research into the tools and concepts involved in our project, and creating documentation around the project. The team created a project report including the spike stories, problem statement, and background/related work. We also researched and organized our findings involving sentiment analysis, topic modeling, data formatting, and bias detection/types of bias in writing. The team also researched the tools needed to build a model for sentiment analysis of a document. We presented these ideas to our client and were able to build a conceptual model of the entire project for which we can develop software.

What went right: Our time management was on track this entire sprint. We had information to present at every meeting and our planning was complete by the end of the sprint, including necessary documentation. We met with our client every week and kept them up to date on our work. The communication went well throughout sprint 1.

Lessons Learned/Improvement: Our conceptual model was at a different scale and understanding than what we have finalized in Sprint 2. Our client and the team developed the model further and changed several misunderstood parts of the model to more fit the client's idea. We realized this late into sprint 1 and have been working to keep our project in-line with the client's wishes. We learned how to better understand the client's requests as well as how to transform ideas from our discussions into tangible requirements.

Sprint-2 Planning: The backlog for sprint 2 consisted of create/submit project report solution approach, create/submit sprint 2 report, create/submit project presentation video, research and explore SpaCy, get data to use, troubleshoot our web scraping tool, explore pattern and Textblob for sentiment analysis, explore VADER for sentiment analysis, explore Gensim topic modeling, write individual task detail reports, explore localized sentiment analysis of proper nouns, and fix/update the project report.
A good portion of this sprint was chosen to start getting hands-on experience with different tools to see how they could help us. In this sprint SpaCy, VADER, Textblob, and Gensim. Using these tools we got a good idea of what the team plans to use going forward for our final model. Outside of exploring different tools most of the other tasks revolved around making sure that the project has good documentation so that others have a clear way of reading up on the project and seeing what it entails. There were a couple tasks such as getting data to use and troubleshooting web scraping tools that are in the category of model testing prep.

Sprint Task Assignments: Creating/submitting project presentation video will be a team effort, with each of us breaking it down into smaller tasks to each complete on our own. Tasks

consist of writing the script for the project problem statement, requirements (functional & non-functional), literature survey and existing solutions, solution approach, current progress, future work/milestones, and recording a demo of the project. Each of us were tasked with researching and exploring various tools for sentiment analysis. Spacy was assigned to William, Vader to Deven, Textblob to Gabe. Then we expanded to various tools pertaining to the project's functionality. Deven explored Gensim for topic modeling. Gabe explored Textblob part-of-speech tagging and developed a localized sentiment analysis tool for finding and tagging proper nouns within text. William explored Beautiful Soup for web scraping and Pandas for managing the .CSV.

Assigned Tasks

William: Research and Explore spaCy, Get data to use, and Troubleshoot web scraping tool.

Gabriel: Gather first data set for sentiment analysis models, explore Textblob as a sentiment analysis tool for news articles, develop sentiment analysis tools for localized analysis of certain topics, and research and development of topic modeling tool for our software.

Deven: Explored VADER as a sentiment analysis tool and explored Gensim as a topic modeling tool.

Sprint Task Details

William Hiatt Sprint Task Details

Task: Research and Explore spaCy
Implementation: Followed an online article on how to do sentiment analysis with spaCy. This article included a way to put website URL's into a .CSV file and then run the model on the CSV file. This article also included textBlob (for sentiment analysis), Beautiful Soup(for web scraping), and Pandas (for managing CSV files) as tools.

Task: Get data to use
Implementation: Iain provided us a .CSV of URLs that include the topic of military required vaccines. This file will be used in conjunction with Pandas to send the data through the web scraper BeautifulSoup. This will then give us the data we are wanting.

Task: Troubleshoot web scraping tool
Implementation: Currently in progress. There is an issue currently with the implementation of Beautiful Soup. This is currently being troubleshooted so we can run the new data through the model. This is a priority that will hopefully be resolved soon.

Task: Write Project Report, Sprint 2 Documentation, and Project Presentation
Implementation: Helped develop the project report-solution approach section. Also helped create the project presentation and presented it in the video. Created and worked on sprint report 2 with the team.

Gabriel Sams Sprint Task Details
Task: Gather first data set for sentiment analysis models.
Implementation: We found a set of data for unit testing our sentiment analysis models, which is a free massive data set of articles annotated with spans of bias within the text. We also gathered a list of URLs containing articles that our client provided for us to use as the first working dataset for the model.

Task: Explore Textblob as a sentiment analysis tool for news articles.
Implementation: I developed multiple sentiment analysis models using the Textblob library for analyzing biased text within a news document. I created a trainable model using Textblob's classifiers that works on single blocks of text, as well as JSON files of text in our dataset. I also created two pre-trained models based on the Pattern and NaiveBayes analyzers within Textblob. These also worked on a block of text and the dataset, and are scalable to work with the working dataset we plan to use (given by our client).

Task: Develop Sentiment Analysis tools for localized analysis of certain topics.
Implementation: Using Textblob, I developed a part-of-speech tagging tool that can recognize proper nouns or first name + last name pairs of proper nouns and gather a localized sentiment based on the analyzed words around them.

Task: Research and development of topic modeling tool for our software.
Incomplete: We are currently in a mix of research and development phases on topic modeling–I am doing research into Gensim and how to implement it into our current model, and Deven has been developing a working prototype topic modeling tool. This will be our main goal in Sprint 3 to develop this tool and merge it with our other tools so we plan on developing it through the start of Sprint 3.

Task: Write Project Report, Sprint 2 Documentation, and Project Presentation
Implementation: Helped develop the project report-solution approach section. Also helped create the project presentation and presented it in the video. Created and worked on sprint report 2 with the team.

Deven Biehler Sprint Task Details
Task: Explored VADER as a sentiment analysis tool
Implementation: VADER is a Python package built to analyze sentences for a sentiment score. I worked through VADER's tutorials to set sentiment analysis for various sentences. The sentiment analysis was based on a scale from positive to negative, with neutral in between. I used this tool to analyze many sentences from news articles, receiving very mixed results. Overall, the tool was easy to use but I felt that it was too simplistic for what we were looking to accomplish.

Task: Explored Gensim as a topic modeling tool
Implementation: Gensim is a Python package built to extract topics from an article. The topics are chosen based on the frequency of the words in the article. I worked through Gensim's

tutorials and found that it worked very well on a news article I inserted. Given a sentence or fragment, I was able to receive a similarity score for many sentences within the article based on the topics within.

Task: Write Project Report, Sprint 2 Documentation, and Project Presentation
Implementation: Helped develop the project report-solution approach section. Also helped create the project presentation and presented it in the video. Created and worked on sprint report 2 with the team.