Project Progress

on

Brand's Sentiment Analysis

Due: 2022-11-14

CS 410: Text Information Systems

Fall 2022

by

ChengXiang Zhai
Professor
Department of Computer Science
University of Illinois at Urbana-Champaign

Submitted By

Ayush Khanna
akhanna6@illinois.edu

Ethan Alberto
ethanma3@illinois.edu

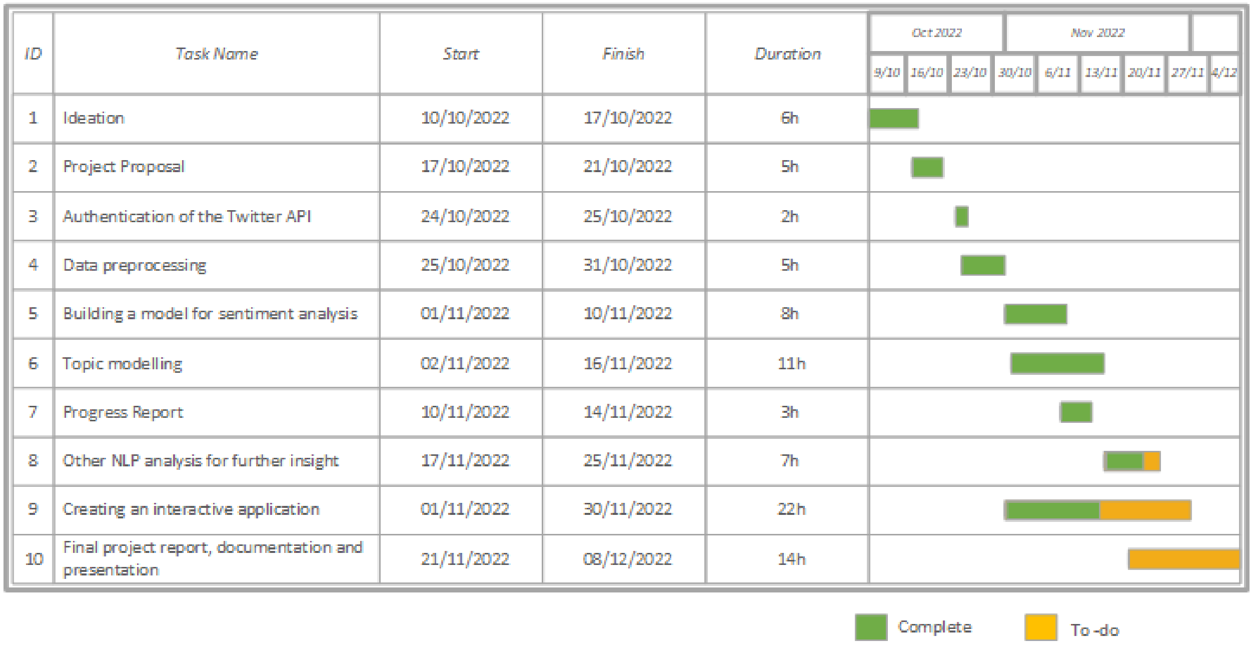
Jayanth Chandra
jc101@illinois.edu

Contents

1 Progress

We are currently on track to deliver everything we initially specified as part part of our Project proposal. We have broken down all the proposed components and divided them among the team. Development of the functionality as well as the dashboard are being done in parallel. So far, we have developed modules on following:

- Tweet pre-processing using snscrape.
- Sentiment analysis on obtained tweets.
- Emotion analysis On Tweets. We are working to improve it's performance now.
- Find most trending topics.
- Find the Satisfaction Score based on Tweets Sentiment.
- Net Promoter Score (NPS) to quantify the the overall sentiment.
- Geo-location mapping for average and poor sentiment tweets.
- Capturing other aspects of the tweets such as most mentions and retweets.
- First layout of our Streamlit Dashboard.



ID	*Task Name*	*Start*	*Finish*	*Duration*
1	Ideation	10/10/2022	17/10/2022	6h
2	Project Proposal	17/10/2022	21/10/2022	5h
3	Authentication of the Twitter API	24/10/2022	25/10/2022	2h
4	Data preprocessing	25/10/2022	31/10/2022	5h
5	Building a model for sentiment analysis	01/11/2022	10/11/2022	8h
6	Topic modelling	02/11/2022	16/11/2022	11h
7	Progress Report	10/11/2022	14/11/2022	3h
8	Other NLP analysis for further insight	17/11/2022	25/11/2022	7h
9	Creating an interactive application	01/11/2022	30/11/2022	22h
10	Final project report, documentation and presentation	21/11/2022	08/12/2022	14h

2 Pending Tasks

As we progress further in project, we are currently working on following to finish our project work:

- **Modules Integration into Dashboard**: We are currently working on designing and aligning the dashboard by integrating all modules mentioned above.
- **OkapiBM25 Algorithm**: Based on the feedback received from the project proposal review to showcase some in-course taught algorithm, we plan on incorporating **OkapiBM25** algorithm to retrieve the most relevant tweets for the most common phrases tweeted about in our dashboard. The work of this part is in progress and we are on track to finish it within the given stipulated time.
- **Incorporate NPS Trend**: NPS serves as a good metric for identifying the overall sentiment of a brands tweets. Although, we have the functionality to calculate the NPS score, we would like to have a view of the historical trend of a particular brand over time. We plan on complimenting the results of the Twitter API with a dataset of scraped tweets with at least 3 months of historical data to calculate the NPS score over time.
- **Evaluate Results in Dashboard**: We are also working on showing the Top 10 Tweets based on the sentiments and their satisfaction score in our Dashboard in order to provide the evaluation criteria of our results. A similar question was also asked in project proposal to provide some review criteria for our results. We believe directly showing the top tweets are the best way to proof the accuracy of our results.
- **Project Testing and Report preparation**: Every project is successful if we have done enough testing and have documented the contents well enough. We need to start the work on our final Project report preparation, decide how to layout the report content, demo on showing the results, etc. We also plan to further optimise and clean up the the code base, documenting the modules and how they can be used. We plan to finish everything as per our project timelines.

3 Challenges

During the course of working on this project, we have faced some challenges which are worth mentioning here:

- **Tweets Library**: Initially, we started capturing the tweets using **tweepy.cursor** api to return the number of tweets from Twitter. But, the problem with this method is that it automatically picks the tweets only for the past 7 days, we don't have any control to expand this date range. So, we decided to use Twitter's **snscrape module**, which gives us an option to expand our query search between any start and end Date range.
- **Tweets Pre-Processing**: We realized that when we pull thousands of tweets by going back to several months, the performance of pulling so many tweets is slow. On an avg. it takes around 10-30 secs to process 1000 tweets from past 7 days. Keeping the performance stat in mind, we decided to limit the tweets range to go back up till past 3 months with max. 5000 tweets, which ever comes first. We also introduced the Streamlit Cache (@st.cache) to improve the performance. This helped us in improving the performance and finish end to end processing of 5000 tweets within 2 mins.

- **Finding Emotions Performance:** When reviewing the Tweets' emotion results and the overall performance of getting the emotion, we realized that the *text2emotion* library takes a good amount of time in returning the results. We are currently working on this challenge to improve the overall performance of this module.

- **Performance of Geo Locations:** Similar to emotion analysis, finding Geo Locations (Longitude and Latitude values) using geopy.geocoders *Nominatim* library takes some time. To overcome the challenge, we decided to show the results of only Negative or Neutral Tweets in our Map results. We are currently working on capturing it's performance and how we can enhance it further.

- **Streamlit Design challenges:** We realized that there are some limitations of streamlit while styling different widgets. In Streamlit, the column class indicator remains same in the dashboard for all *st.columns* classes having equal number of columns. This results into same styling being applied to all other widgets with similar number of columns. So, we spent sometime in figuring out the final layout of our dashboard.