



Project Number  
S16-036

Team  
Julian Esteban, Kanav Tahilramani, Matthew Chatten, Sujay Bandarpalle

Advisor  
Dr. Shantenu Jha

May 2, 2016

Submitted in partial fulfillment of the requirements for senior design project

**Electrical and Computer Engineering Department**  
**Rutgers University, Piscataway, NJ 08854**

## Abstract

The goal of this project is building a webapp providing analytical analysis of online social media behavior. Specifically, due to the open nature of its data and a public API, the site Reddit was chosen as a focus of the project. Reddit is a semi-anonymous public platform; users are identified by self-chosen usernames, and can post comments or links to a wide variety of separate communities called “subreddits”. It is these two key elements – users and subreddits – which this project focuses on. The RedReaper tool takes as input a username or the name of a subreddit, and outputs a wide-ranging analysis of it.

Three key categories of analysis and data are produced by this tool. The first category is general statistical data, such as number of submissions, popularity of comments, details of posting history, etc. This is accomplished through fairly simple techniques - accessing the appropriate dataset and counting up the relevant values. The second and most important category is content analysis. This includes calculating the complexity of language used, detecting personally identifiable information, and determining the positive or negative sentiment of posts. The final category is metadata analysis, focused on revealing how variables like post time, post score, and comment length relate to each other.

The aim of providing all this analysis is to give the user of the webapp an understanding both of the individual Reddit user or subreddit and of just how much data and personal details can be extracted and compiled from public social media sites. It combines the personal interest of seeing information on how you or other people speak with the educative experience of seeing what natural language processing and metadata can reveal.

## Table of Contents

|        |   |    |
|--------|---|----|
| 1.     | INTRODUCTION .....                                  | 4  |
| 1.1.   | <i>Background Information</i> .....                 | 4  |
| 1.2.   | <i>State of the Art and Relevant Projects</i> ..... | 4  |
| 1.3.   | <i>Problem Addressed by Project</i> .....           | 5  |
| 1.3.1. | Narrowly Defined Objective .....                    | 5  |
| 1.3.2. | Broadly Defined Objective .....                     | 6  |
| 1.4.   | <i>The Adopted Approach</i> .....                   | 6  |
| 2.     | APPROACH TO THE PROBLEM .....                       | 7  |
| 2.1.   | <i>Methods</i> .....                                | 7  |
| 2.1.1. | Basic Project Framework .....                       | 7  |
| 2.1.2. | Implementation Challenges .....                     | 7  |
| 2.1.3. | Time Constraints and Their Impact .....             | 8  |
| 2.1.4. | Required Knowledge Base and Tools for Project ..... | 8  |
| 2.2.   | <i>Work Done by Team Members</i> .....              | 9  |
| 2.2.1. | Stage 1 .....                                       | 9  |
| 2.2.2. | Stage 2 .....                                       | 9  |
| 2.2.3. | Stage 3 .....                                       | 9  |
| 2.3.   | <i>Use of Standards</i> .....                       | 10 |
| 3.     | PRODUCT RESULTS .....                               | 11 |
| 3.1.   | <i>Back-end</i> .....                               | 11 |
| 3.2.   | <i>Front-end</i> .....                              | 12 |
| 3.2.1. | The Main Page .....                                 | 12 |
| 3.2.2. | User Analysis .....                                 | 13 |
| 3.2.3. | Subreddit Analysis .....                            | 18 |
| 4.     | DISCUSSION .....                                    | 20 |
| 4.1.   | <i>Economic Cost and Impact</i> .....               | 20 |
| 4.2.   | <i>Environmental Impact</i> .....                   | 20 |
| 4.3.   | <i>Social Impact</i> .....                          | 20 |
| 4.4.   | <i>Scalability</i> .....                            | 21 |
| 5.     | CONCLUSIONS .....                                   | 23 |
| 6.     | REFERENCES .....                                    | 24 |
| 7.     | APPENDIX .....                                      | 26 |
| 7.1.   | <i>Use Case Discussion</i> .....                    | 26 |
| 7.2.   | <i>Gantt Chart</i> .....                            | 31 |

# 1. Introduction

The following sections explain the problem Redreaper aims to address, and the overall context in which the project was begun.

## 1.1. Background Information

Reddit is an independent social media site, owned by “reddit Inc.”, which calls itself “the front page of the internet.”[1] It is essentially a network of discussion pages, organized by topic, called “subreddits” in which users can post links or discussion topics for other users to see, comment on, and vote on. The key to the website is its “karma” system, in which users can “upvote” or “downvote” content, both submissions and comments, resulting in community-driven “front pages” of curated content. With a weighting algorithm that heavily favors new content over old, popular submissions will eventually drop off of the front of a subreddit to be replaced by new material.

Another factor that makes Reddit relatively unique among social sites, and which led to its choice as the subject of this project, is that almost all subreddits are completely public, and moreover all user histories are public. While Reddit can simply be browsed, to vote on or submit content a user account with a unique username must be created. These usernames are almost never the users’ real names, but they are still trackable and associated with only one person. Given the open nature of such data and the existence of a solid API for accessing it, Reddit is the perfect test case for analyzing online social behavior in aggregate.

## 1.2. State of the Art and Relevant Projects

Because Reddit is so ideal for this type of data mining and analysis, this project is not the first to attempt the task. There are a host of websites and services dedicated to Reddit data – for example, FiveThirtyEight has made a “Reddit Ngram” available.[2] (It displays frequency of word or phrase use within a corpus of Reddit comments from 2007 to 2015, much like the Google “Ngram” does for Google’s collection of digitized books.) Other notable examples include KarmaDecay[3], a website which can perform reverse image searches on Reddit, and Karmawhores[4], which tracks the top users of Reddit by their karma scores. However, none of these do any analysis of individual users or subreddits, which is the goal of this project.

There are three existing websites which perform some sort of analysis on either users or subreddits. The first is roadtolarissa’s “Reddit Comment Visualizer”[5], which can graph a user’s comments by number, length, or karma according to subreddit over time. This is useful and fairly well-displayed data, but it is

also simply metadata. The most you can do is see general commenting trends. This project is aimed more at specific pieces of data and analysis which go deeper into the actual content of a user's history.

The second is Redective[6], which retrieves metadata for both users and subreddits upon request. However, the output is very limited, and not displayed in a particularly appealing or even understandable manner. It displays a small amount of basic information, counts submissions by subreddit, lists the most frequent words used, and displays the top scoring submissions. Our project aims to be more user-friendly, include more detailed information, and focus more on detailed content analysis rather than simple metadata.

The third and final similar work in this field is the more impressive SnoopSnoo[7], which again allows for analysis of both users and subreddits, though for subreddits only very basic information is presented. The user analysis includes a solid selection of basic data, such as best and worst comments, as well as a good selection of metadata like posts by day, posts by time, and posts by subreddit organized and displayed in visually appealing graphs. Unlike every other work mentioned so far, SnoopSnoo actually tries to extract some information from user's comments using natural language processing, displaying items such as what the user has explicitly said they like or the user's stated religious beliefs, whenever available. This is closer to what we hope to accomplish with our own project, but it is still limited. We intend to do deeper and wider natural language processing and content analysis, along with an even better selection of metadata and regular data analysis. We also intend to provide much more analysis for subreddits.

### **1.3. Problem Addressed by Project**

The core problem addressed by RedReaper can be looked at narrowly or broadly. In a narrow sense, Redreaper's goal is to provide meaningful analysis and data on Reddit users and subreddits. In a broad sense, Redreaper's goal is to give its users an understanding how social media data in aggregate can reveal more than one might expect.

#### **1.3.1. Narrowly Defined Objective**

As discussed in the section on the State of the Art, existing services using Reddit data do not match up to the wide space of possibilities in natural language processing or even the extensive metadata Reddit provides. As existing solutions to the problem of "I want to see analysis and data on a user or subreddit." go, there is significant room for improvement, and that is where Redreaper comes in. Redreaper will

provide broader and deeper analysis for both users and subreddits, by collecting more metadata, by doing more in-depth natural language processing, and by presenting this data in an understandable manner.

### 1.3.2. Broadly Defined Objective

This is the era of both social media and Big Data, and yet most regular users of social media don't have a good grasp on quite what that means. Social media companies themselves, such as Facebook, are constantly sifting through and analyzing the data available to them on their users, but users themselves don't have such an opportunity. Because Reddit is an open, public site with a robust API, Redreap can give its users a taste of what is currently possible in social media analysis. Especially important is the natural language processing component, which can uncover personal information, personal habits and proclivities, and more, but whose power is generally unknown to the public. This lack of understanding can be rectified by Redreap, by showing just how much can be revealed by automatic analysis of social media users and communities.

## 1.4. The Adopted Approach

The basic approach is to build a website that meets the defined objectives. The problem is best solved by a website because, at its core, Redreaper is built to meet the requests of users. Users of the site request analysis of either Reddit users, by their Reddit username, or of Reddit subreddits, by the subreddit's name. Because actually analyzing these inputs requires a large amount of computation time, and because there are over 36 million Reddit users and over 850 thousand subreddits[8], the results cannot all be precomputed. Thus, the output depends on the user's input and on having access to Reddit through its API, making a website approach ideal.

## 2. Approach to the Problem

The following sections detail the work done to complete the objectives of the project, the standards and guidelines followed during that process, and the results of that work.

### 2.1. Methods

The nature of the project problem meant that a software solution was required. Therefore, the execution of the project has centered entirely on programming goals and challenges.

#### 2.1.1. Basic Project Framework

Any website requires some sort of “programming stack” that will handle the front-end and back-end operations necessary. For this project, the MEAN stack[9] was chosen, which includes mongoDB, express, AngularJS, and NodeJS. It was chosen for its ease of use, and because of previous experience with the framework. For coordination of the project coding, Github[10] was chosen and a repository was made with all group members included. This allowed for effective version control, and made it easy for individual group members to make their own branches of the project to implement features or test out new ideas. The project is designed such that it can run on any computer with the proper software installed – the software required is included in the repository and a few simple commands will install it – so development can be done on any device. The project can also then be deployed wherever is best at the moment, with an eye towards eventually having it run on its own server.

#### 2.1.2. Implementation Challenges

Key implementation challenges first included setting up the full stack of software such that a website would actually load that could be interacted with. This part of the project took up much of the initial weeks, focusing on authenticating API calls to Reddit, storing and retrieving data from the database correctly, and integrating a variety of tools. Most notably, integrating NLP tools proved challenging, since they had their own dependencies. There was also considerable work done to ensure that the program would run on a variety of setups.

The second major challenge area is in performance. Analysis has two major chokepoints: API calls, and NLP processing. Reddit’s API only allows for up to two API calls per second, and limits the size of each API call fairly strictly, so a user with an extensive history can require over 20 API calls in order to gather all the necessary data. Furthermore, the “tokenization” process involved in NLP analysis can take a lot of computing power and time, especially if the user in question tends to write longer comments or submissions. These problems were tackled by setting up a complex authentication system for API calls which allows for an increased number of calls per second, by separating the API calls from the main

thread of the program so that other tasks can proceed while Redreaper waits for more response from Reddit, and by optimizing the NLP analysis done by the program.

A third major area of challenge was general debugging and code cleanliness, troubles common to all software projects but still notable nonetheless. Memorable bugs include one that caused analysis of users to run twice, drastically increasing response times, and edge cases in NLP or API responses that caused unexpected results or errors. Working with a four person team also meant taking special care to document implemented features and keep a structure to the overall project that allowed for simple additions of new features or editing of old ones. This meant taking care to comment code, and at one point taking the time to reorganize the main analysis code in order to make it more modular.

### 2.1.3. Time Constraints and Their Impact

There is a very wide variety of features we could implement in our project. Because the end goal is simply informative analysis of Reddit users and subreddits, the actual implementation of that analysis can take many different shapes. In the beginning stages of the project the team produced multiple pages of various ideas that could be implemented, knowing that we would have to prioritize and choose only those that were feasible and most valuable. And indeed, there are many ideas that will be left unimplemented.

The other key step which has been prevented by time constraints is testing a full deployment of the project online. Releasing a public Redreaper website would mean serious bug testing, as there would almost undoubtedly be new issues either discovered by the influx of users or caused by them. It's difficult to judge how much interest there would be ahead of time, but there's a chance that server capacity would be overwhelmed since analysis is computationally expensive.

### 2.1.4. Required Knowledge Base and Tools for Project

Because this is a software project where all team members need to contribute code, the key knowledge required is basic programming practices. How to perform basic actions like creating functions, how to install various sets of software, how to operate in a coding environment with databases, websites, and API calls, and more are all required. Furthermore, this project also meant learning new programming tools, like NLP or various Javascript frameworks. The exact amount of learning varied from team member to team member, but all had to do a fair bit of study in order to execute the project.



## **2.2. Work Done by Team Members**

The project can essentially be divided into three main stages: the startup stage, the feature implementation stage, and the project finalization stage.

### **2.2.1. Stage 1**

In the first stage, the entire team worked together to produce proposals, planning documents like the list of possible types of analysis, etc. This also extended to the early stages of the project where setting up a working programming stack and website framework to implement features on was the goal. During this stage the team generally operated by setting up specific meeting times where everyone would join a Skype call and discuss the issue of the day, either troubleshooting problems, discussing ideas, or working together on some sort of programming solution. Kanav Tahilramani's contributions here, as the team member most experienced with MEAN, were especially valuable.

### **2.2.2. Stage 2**

The second stage of the project was the feature stage, where each team member focused in implementing specific features of the project. Julian Esteban took the lead on the project's centerpiece, NLP analysis features, and also worked on overall performance. Kanav Tahilramani focused on overall performance, fixing bugs, and NLP design. Matthew Chatten focused on data collection and metadata analysis. Sujay Bandarpalle focused on the front-end display of the results, and went above and beyond to develop logos, gifs, interactive graphs, and more for a better user experience. There was, of course, plenty of communication between team members on what each person was implementing and how to solve shared problems, but on the whole work was done separately according to individual interests and strengths.

### **2.2.3. Stage 3**

The third and final stage of the project was finalization, where final features were agreed upon, a presentation was prepared and delivered, and project deliverables such as the poster, this report, and a video were produced. Final features were chosen by group consensus, but completed in the usual manner. These included things like overall sentiment analysis for subreddits and details on user's top five and bottom five posts. The poster and report were mostly made by Matthew Chatten, but with group input, and the presentation and video were full group efforts. In the end, considering the entire scope of the project, every member of the group made valuable and essential contributions to the final project.

## 2.3. Use of Standards

- a. Reddit API
  - a. Documented at <https://www.reddit.com/dev/api>
  - b. Required for retrieving data necessary for analysis
  - c. OAuth2[12] – SSL protocol for API authentication
- b. Stanford NLP
  - a. Documented at <http://nlp.stanford.edu/software/>
  - b. Selection of open source software for NLP analysis
  - c. Project use centered on CoreNLP[13]
- c. MEAN Stack
  - a. Documented at mean.io
  - b. Programming stack used for making responsive websites

### 3. Product Results

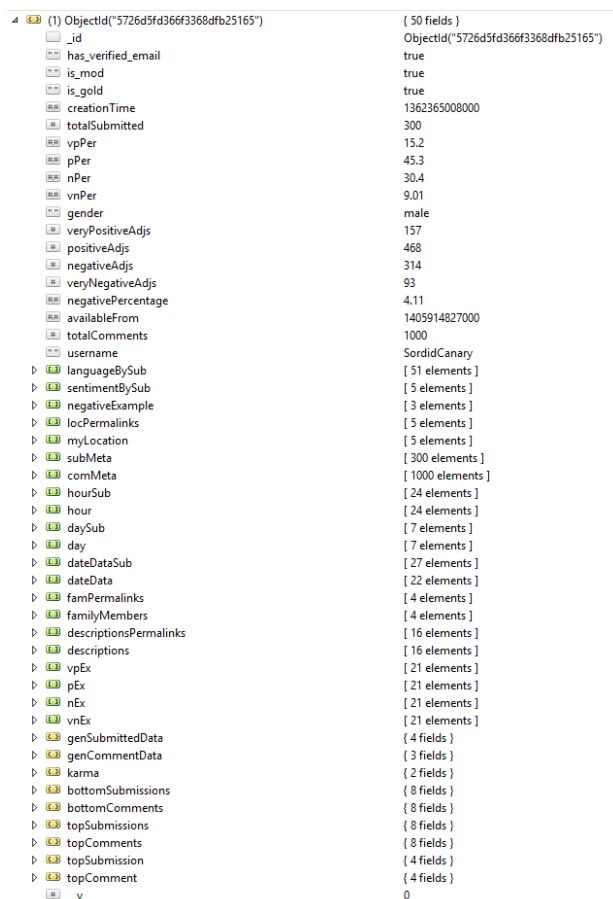
The following sections detail the final product: both the back-end and the front-end.

#### 3.1. Back-end

The full codebase for Redreaper is publicly available on Github[10]. In total, Redreaper is made up of thousands of lines of code assembled over months across dozens of files. For core user analysis, the files “reddit.controller.js” in /server/api/reddit/ and ‘user.model.js’ in /server/api/user/ should be looked into. The former has functions for making API calls and analyzing the returned data, while the latter defines the database model for users. Similar files exist for subreddit analysis.

Once analysis is complete, everything that has been found will be stored in the database in an entry associated with the requested user or subreddit, organized by type of information and labelled appropriately.

The below is an image of an example user’s database entry, “/u/SordidCanary”, where you can see basic account details at the top and arrays or objects filled with metadata and NLP results at the bottom.

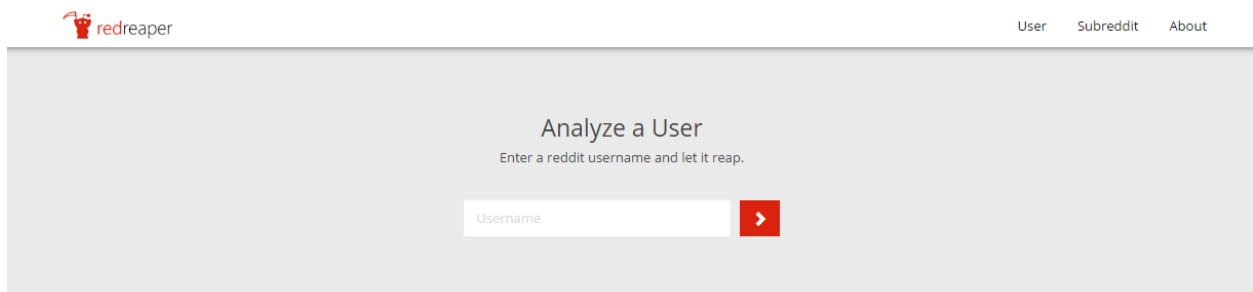


|  |                                      |
|--|--------------------------------------|
| (1) ObjectId("5726d5fd366f3368dfb25165") | { 50 fields }                        |
| _id                                      | ObjectId("5726d5fd366f3368dfb25165") |
| has_verified_email                       | true                                 |
| is_mod                                   | true                                 |
| is_gold                                  | true                                 |
| creationTime                             | 1362365008000                        |
| totalSubmitted                           | 300                                  |
| vpPer                                    | 15.2                                 |
| pPer                                     | 45.3                                 |
| nPer                                     | 30.4                                 |
| vnPer                                    | 9.01                                 |
| gender                                   | male                                 |
| veryPositiveAdjs                         | 157                                  |
| positiveAdjs                             | 468                                  |
| negativeAdjs                             | 314                                  |
| veryNegativeAdjs                         | 93                                   |
| negativePercentage                       | 4.11                                 |
| availableFrom                            | 1405914827000                        |
| totalComments                            | 1000                                 |
| username                                 | SordidCanary                         |
| languageBySub                            | [ 51 elements ]                      |
| sentimentBySub                           | [ 5 elements ]                       |
| negativeExample                          | [ 3 elements ]                       |
| locPermalinks                            | [ 5 elements ]                       |
| myLocation                               | [ 5 elements ]                       |
| subMeta                                  | [ 300 elements ]                     |
| comMeta                                  | [ 1000 elements ]                    |
| hourSub                                  | [ 24 elements ]                      |
| hour                                     | [ 24 elements ]                      |
| daySub                                   | [ 7 elements ]                       |
| day                                      | [ 7 elements ]                       |
| dateDataSub                              | [ 27 elements ]                      |
| dateData                                 | [ 22 elements ]                      |
| famPermalinks                            | [ 4 elements ]                       |
| familyMembers                            | [ 4 elements ]                       |
| descriptionsPermalinks                   | [ 16 elements ]                      |
| descriptions                             | [ 16 elements ]                      |
| vpEx                                     | [ 21 elements ]                      |
| pEx                                      | [ 21 elements ]                      |
| nEx                                      | [ 21 elements ]                      |
| vnEx                                     | [ 21 elements ]                      |
| genSubmittedData                         | { 4 fields }                         |
| genCommentData                           | { 3 fields }                         |
| karma                                    | { 2 fields }                         |
| bottomSubmissions                        | { 8 fields }                         |
| bottomComments                           | { 8 fields }                         |
| topSubmissions                           | { 8 fields }                         |
| topComments                              | { 8 fields }                         |
| topSubmission                            | { 4 fields }                         |
| topComment                               | { 4 fields }                         |
| _v                                       | 0                                    |

## 3.2. Front-end

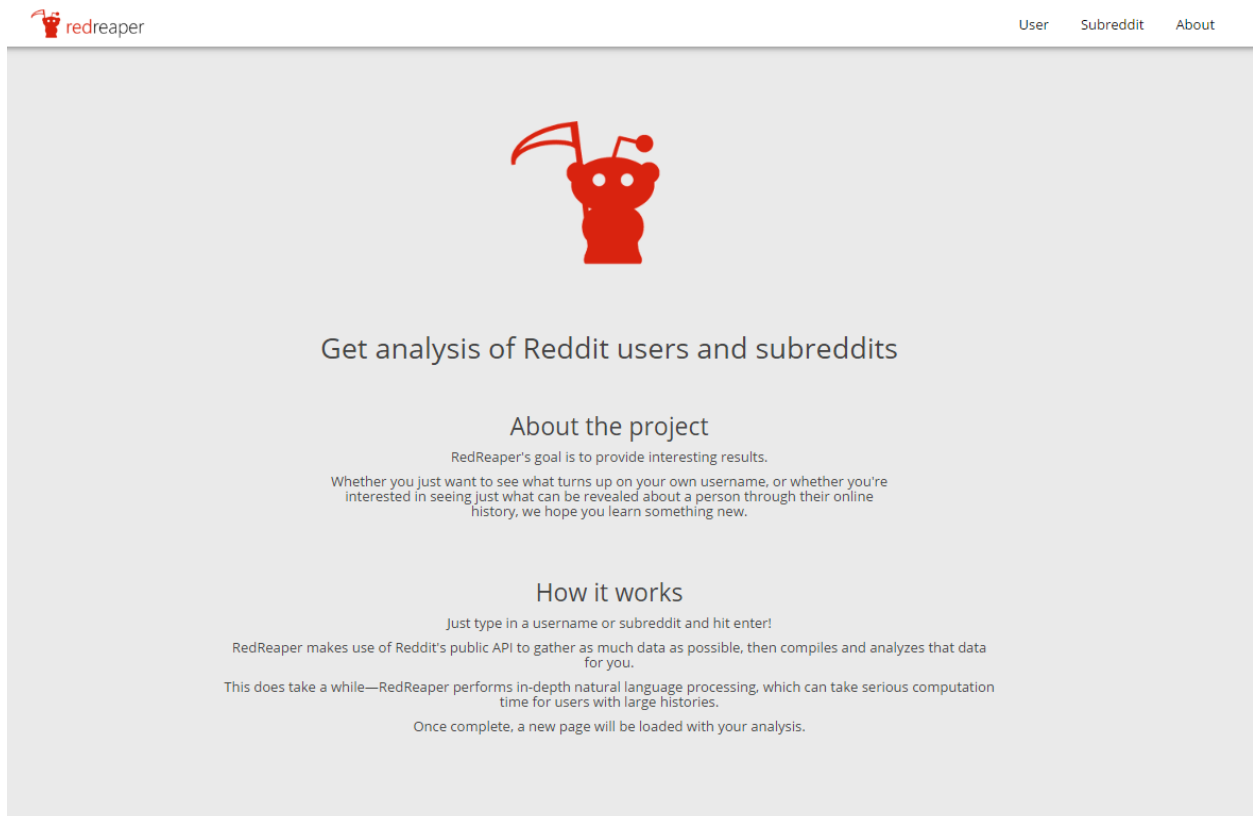
This section explains the results a user of Redreaper would see, using the previous example, “/u/SordidCanary”, and an example subreddit, “/r/politics”.

### 3.2.1. The Main Page



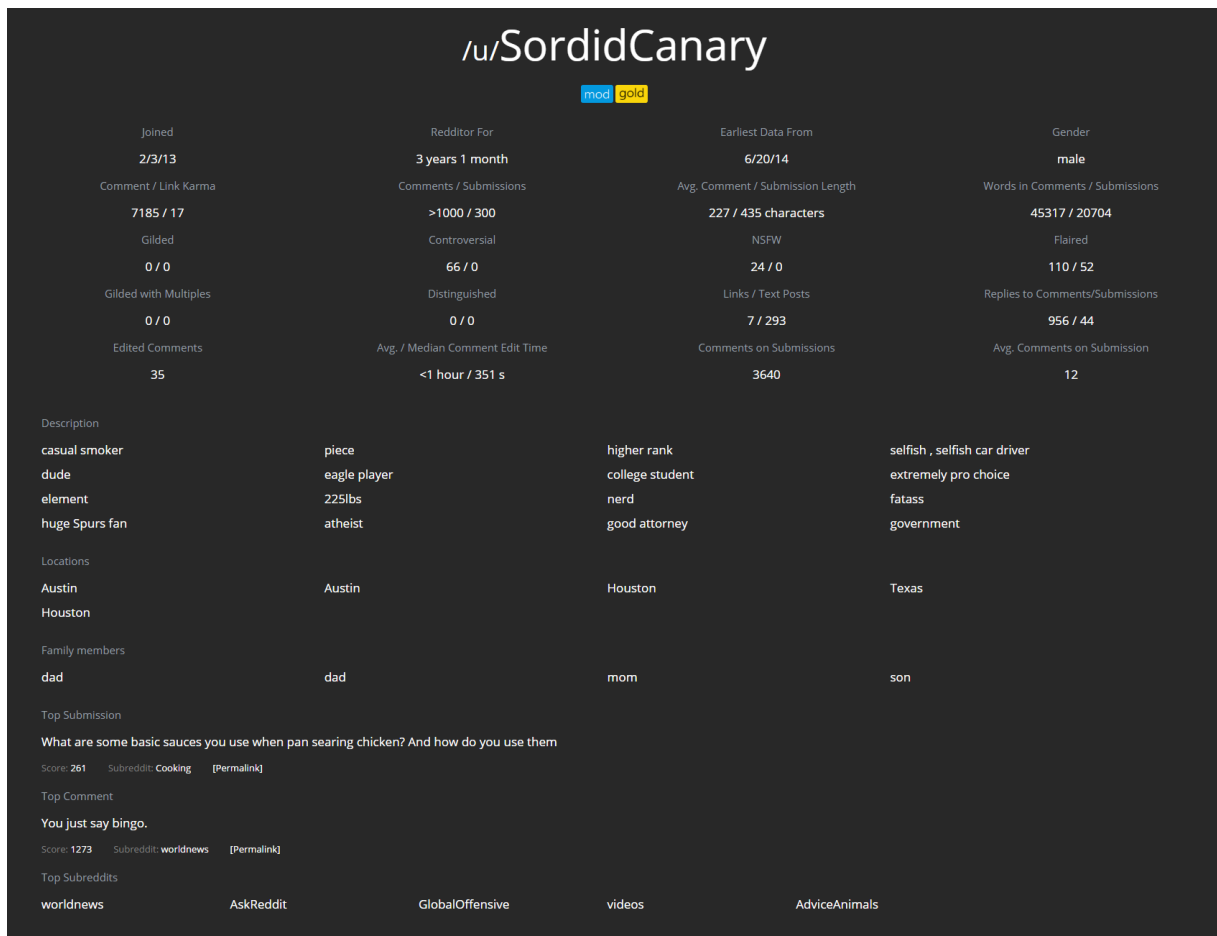
The screenshot shows the top of the RedReaper website. In the top left corner is the RedReaper logo, which consists of a red alien head icon and the text 'redreaper'. In the top right corner are three links: 'User', 'Subreddit', and 'About'. The main content area has a light gray background. It features the heading 'Analyze a User' in a medium-sized font. Below this heading is a smaller line of text: 'Enter a reddit username and let it reap.' Underneath that is a white input field with the placeholder text 'Username'. To the right of the input field is a red button with a white right-pointing arrow.

This is the page that greets the user. A username can be entered, or they can switch to subreddit analysis, or they can go to the about page:



The screenshot shows the main page of the RedReaper website. It has the same header as the previous screenshot, with the RedReaper logo and navigation links. The main content area has a light gray background. At the top center is a large red alien head icon holding a scythe. Below this icon is the heading 'Get analysis of Reddit users and subreddits'. Underneath that is the heading 'About the project'. Below this heading is a paragraph of text: 'RedReaper's goal is to provide interesting results. Whether you just want to see what turns up on your own username, or whether you're interested in seeing just what can be revealed about a person through their online history, we hope you learn something new.' Below this paragraph is the heading 'How it works'. Below this heading is a paragraph of text: 'Just type in a username or subreddit and hit enter! RedReaper makes use of Reddit's public API to gather as much data as possible, then compiles and analyzes that data for you. This does take a while—RedReaper performs in-depth natural language processing, which can take serious computation time for users with large histories. Once complete, a new page will be loaded with your analysis.'

### 3.2.2. User Analysis

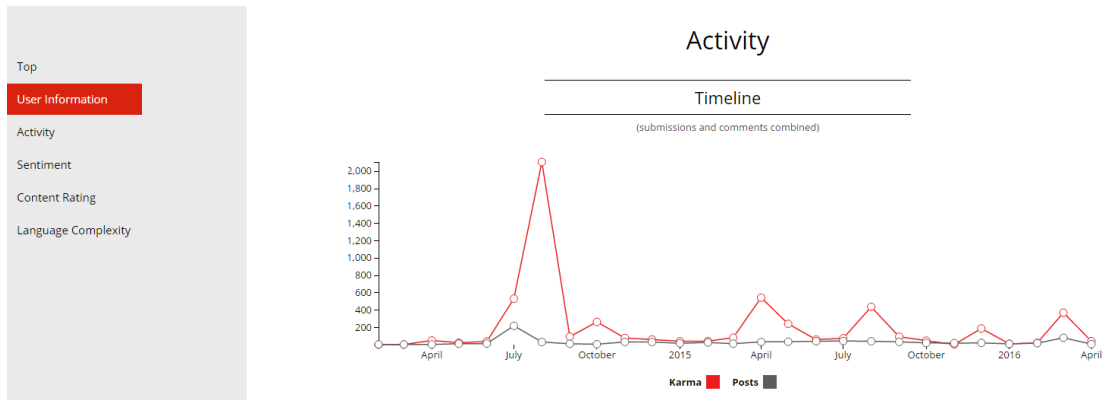


This is the first thing a user sees – a large block giving basic account details, metadata, and specific pieces of personal information extracted through NLP processing.

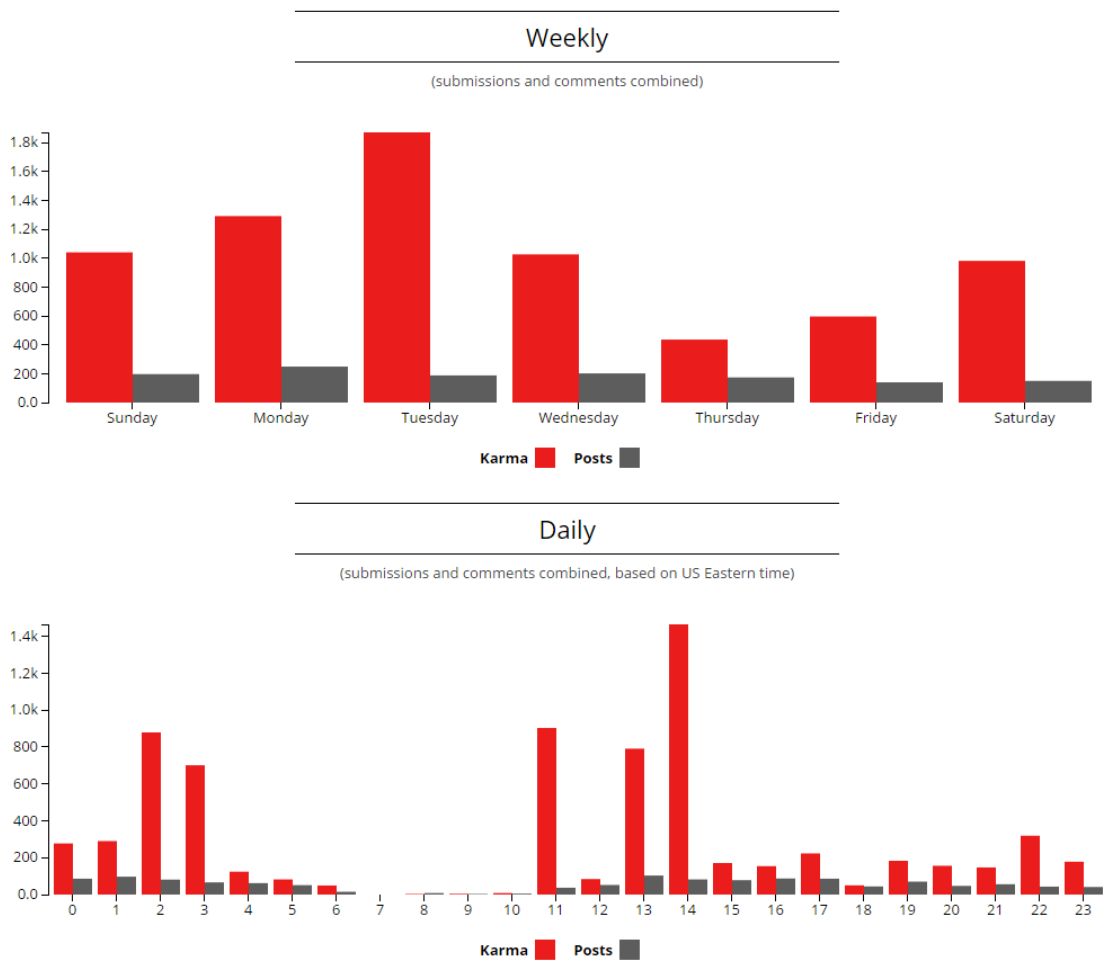
The top segment includes details like how old the account is, the total karma on the account, etc. but also includes more unusual data like the number of comments made which were replies to comments vs. replies to a submission. This can indicate whether or not a user is more likely to participate in discussions or not. It also displays average and median edit times for comments, indicating if a user is going back months later to edit out parts of their comments.

The middle segment includes extracted personally identifiable information or PII. Here we can see that SordidCanary has described himself as an “atheist” and a “college student” living near “Austin” and “Houston”. Each of these pieces of PII can be clicked on in order to go directly to the comment where it was found, so the user can investigate the context and perhaps delete the comment.

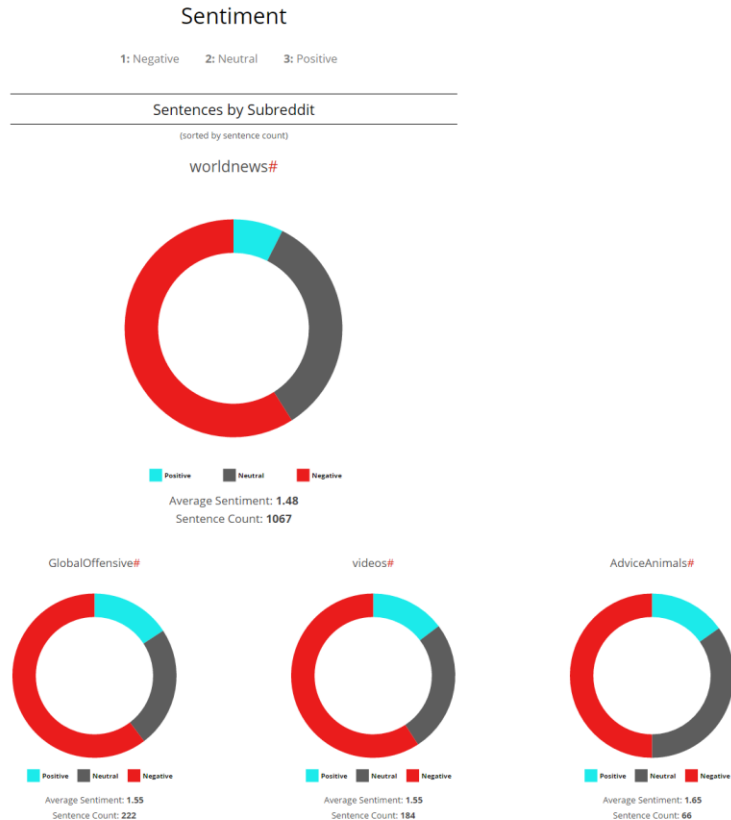
The bottom segment includes the top comment and submission by the user, as well as their most popular subreddits.



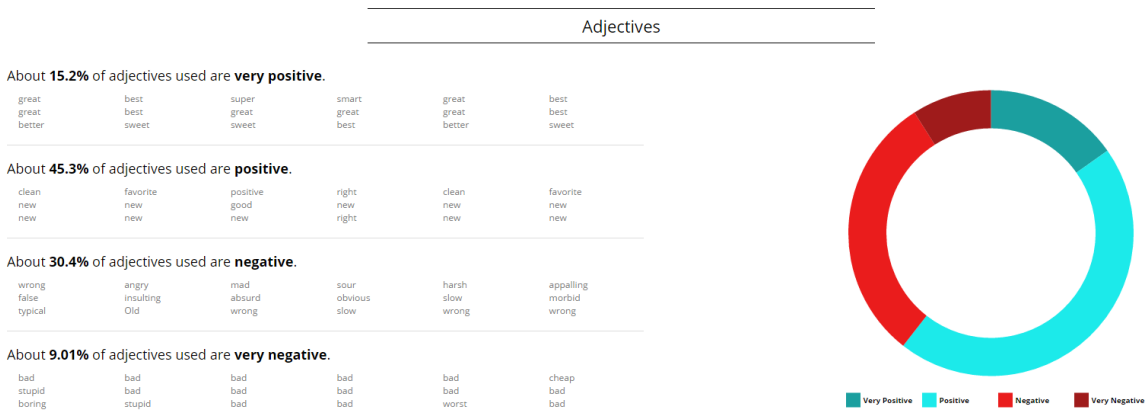
Here we can see a graph of karma and posts over the user's lifetime. The points can be hovered over for exact details. Also included in this image is the lefthand directory, which indicates the current section of analysis the user has scrolled to and which can be clicked on to go to another section.



This is data on what days of the week and what hours of the day the user posts, both by raw count and by karma accumulated. We can see from the second chart that this user likely lives in the US.



This section is “sentiment analysis,” one of the core NLP functions of Redreaper. The top five subreddits each have the user’s comments analyzed for positive, neutral, or negative sentiment. These graphs show the sentiment values for each subreddit, as well as the number of sentences used to make the determinations. We can see that this user, for example, is less positive in /r/worldnews and more positive in /r/AdviceAnimals, a subreddit where funny pictures are posted.



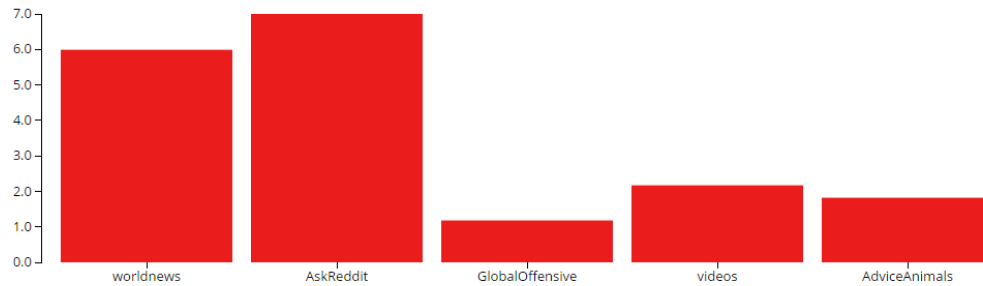
This section shows the overall percentages of non-neutral adjectives used, as well as displays examples of those used. This both offers a view into the user’s writing and shows their general attitude overall.

## Content Rating

### Comment Level

0: Reply to Post    1: Reply to Comments

### Top 5 Subreddit Average Scores



### Submissions

#### Top 5

[\[1\]](#) [\[2\]](#) [\[3\]](#) [\[4\]](#) [\[5\]](#)

Average Sentiment: **1.67**

Average Score: **219**

Average # of Comments: **117**

Average Length: **498**

Average Type of Post: **Text**

#### Bottom 5

[\[1\]](#) [\[2\]](#) [\[3\]](#) [\[4\]](#) [\[5\]](#)

Average Sentiment: **1.5**

Average Score: **0**

Average # of Comments: **10**

Average Length: **173**

Average Type of Post: **Text**

### Comments

#### Top 5

[\[1\]](#) [\[2\]](#) [\[3\]](#) [\[4\]](#) [\[5\]](#)

Average Sentiment: **2.25**

Average Score: **536**

Average Level: **1**

Average Length: **71**

Average Type of Post: **Link**

#### Bottom 5

[\[1\]](#) [\[2\]](#) [\[3\]](#) [\[4\]](#) [\[5\]](#)

Average Sentiment: **1**

Average Score: **-9**

Average Level: **1**

Average Length: **161**

Average Type of Post: **Link**

This section includes a graph of average karma in the user's top five subreddits, so they can see where their content is most well-liked. It also gives details on their top and bottom posts. We can see that for both comments and submissions, SordidCanary's top posts have much more positive sentiment scoring. We can also find interesting results, like the fact that his top submissions are longer while his top comments are shorter.

Also, all of these top and bottom posts are linked, so you can see exactly what was said. SordidCanary's bottom comments are all mean or argumentative, and could be a good candidate for deletion.



## Language Complexity

1: Simple   2: Compound   3: Complex   4: Compound Complex

---

### Top 5 Subreddits

---

#### EDM#

Language Complexity Score: **8.08**

Average Sentence Complexity: **4**

Average Word Length: **4.08**

#### buildapcforme#

Language Complexity Score: **7.73**

Average Sentence Complexity: **1.86**

Average Word Length: **5.87**

#### changemyview#

Language Complexity Score: **7.39**

Average Sentence Complexity: **3.19**

Average Word Length: **4.2**

#### flyings#

Language Complexity Score: **7.36**

Average Sentence Complexity: **3.33**

Average Word Length: **4.03**

#### politics#

Language Complexity Score: **7.24**

Average Sentence Complexity: **2.69**

Average Word Length: **4.55**

---

### Bottom 5 Subreddits

---

#### HomeImprovement#

Language Complexity Score: **5.46**

Average Sentence Complexity: **1.75**

Average Word Length: **3.71**

#### explainlikeimfive#

Language Complexity Score: **5.38**

Average Sentence Complexity: **1.5**

Average Word Length: **3.88**

#### StreetFighter#

Language Complexity Score: **5.35**

Average Sentence Complexity: **1.69**

Average Word Length: **3.66**

#### DarkSouls2#

Language Complexity Score: **5.19**

Average Sentence Complexity: **1**

Average Word Length: **4.19**

#### GlobalOffensiveTrade#

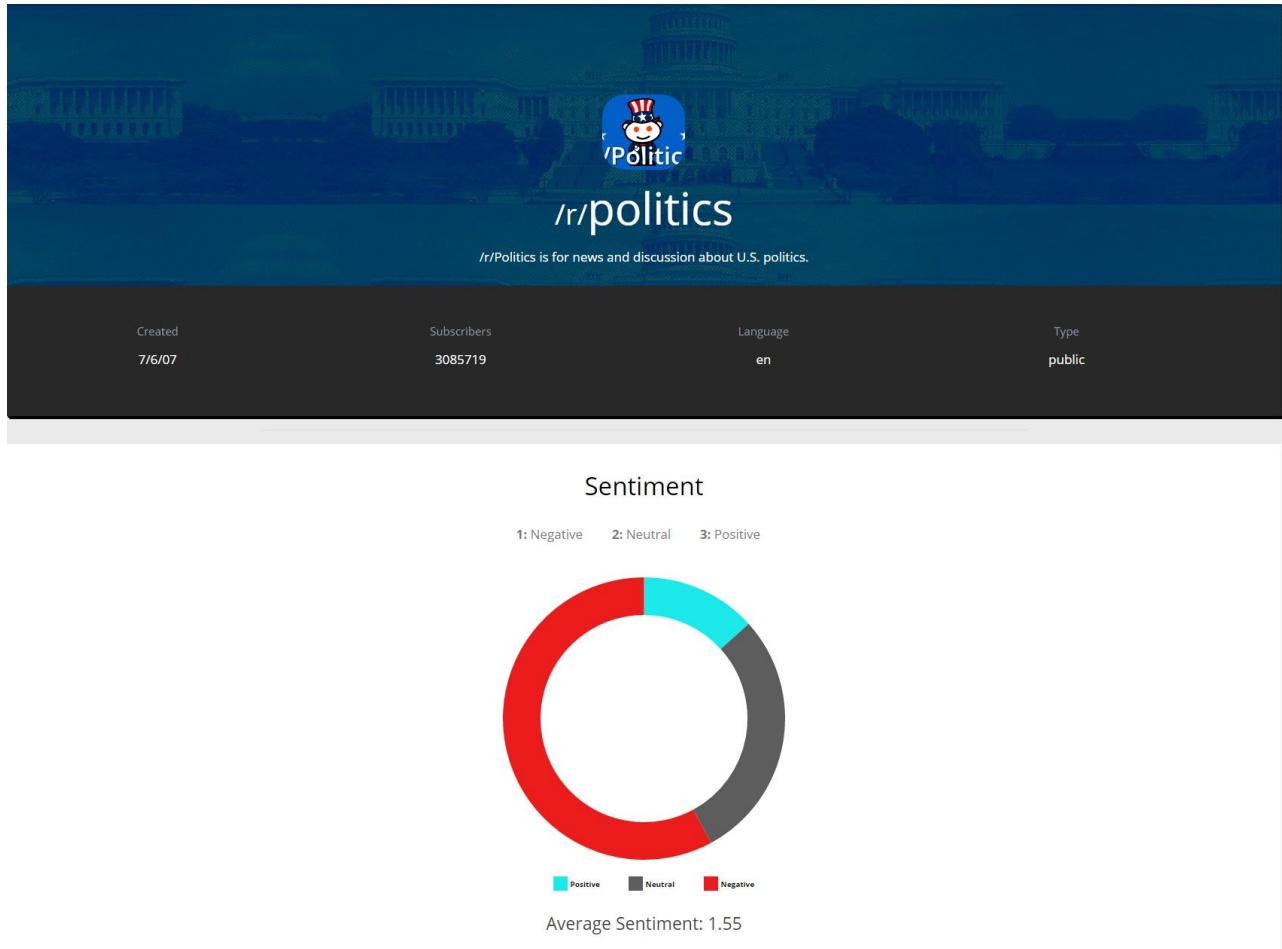
Language Complexity Score: **3.78**

Average Sentence Complexity: **1**

Average Word Length: **2.78**

This final section looks at the language complexity by subreddit, displaying both top and bottom scoring subreddits. It can be seen that SordidCanary uses much more complex language in subreddits like “/r/changemyview”, which is for trying to convince other users of a viewpoint, and much less complex in gaming subreddits like “/r/DarkSouls2”.

### 3.2.3. Subreddit Analysis



Here we can see basic data on the requested subreddit, as well as a graph of overall sentiment within the subreddit. This is a look at the most recent status of the subreddit, taken from comments on current hot posts.

However, the centerpiece of subreddit analysis is sentiment data on specific search terms. For politics, terms relating to current presidential candidates can be chosen. Analysis can then be run multiple times over subsequent days or weeks as the subreddit changes in order to see changes in opinion over time, as well as changes in how much each subject is discussed.

In the following images are two snapshots taken a few days apart. On the whole, sentiment towards each candidate has not changed much, but differences are apparent upon closer inspection. The first snapshot has more positive sentiment for the Democratic candidates, and more negative sentiment for the Republican candidate. The second snapshot has considerably more discussion of the Democratic candidates than the first, and less discussion of the Republican.

## Key Topics

### Presidential Candidates

Hillary Clinton  
Search Terms: "hillary", "clinton"  
Average Sentiment: 1.29  
Sentences mentioned in: 372



Bernie Sanders  
Search Terms: "bernie", "sanders", "bernard"  
Average Sentiment: 1.25  
Sentences mentioned in: 252



Donald Trump  
Search Terms: "donald", "trump"  
Average Sentiment: 1.35  
Sentences mentioned in: 251



## Key Topics

### Presidential Candidates

Hillary Clinton  
Search Terms: "hillary", "clinton"  
Average Sentiment: 1.37  
Sentences mentioned in: 448



Bernie Sanders  
Search Terms: "bernie", "sanders", "bernard"  
Average Sentiment: 1.39  
Sentences mentioned in: 401



Donald Trump  
Search Terms: "donald", "trump"  
Average Sentiment: 1.3  
Sentences mentioned in: 174



## **4. Discussion**

The following sections discuss the various costs and impacts of the project at its current completed stage, and the possibilities of further development.

### **4.1. Economic Cost and Impact**

Development of the project costs only the team members' time at this stage. This is not trivial, considering software developers can generally command a high salary, and many hours of work have been put into the project. At regular hourly wages, the development time cost of the project could conceivably be in the thousands of dollars. Luckily, we did not need to pay ourselves a salary, so the projects actual expenses were zero.

However, the long-term sustainability of Redreaper would not be free. While it has currently only been run on developer computers, for a full deployment a server would need to be rented and a web domain purchased. The actual cost of this would depend on how much traffic the website received; websites can be hosted on rented servers for as little as \$8 dollars a month[14], but due to the computationally intense nature of Redreaper a large userbase might require a beefy server costing as much as \$150 a month or more in the extreme case.

There is also the need to consider continued maintenance on the developer side. Bugs will have to be quashed, and the website will have to be monitored, and there will probably be a need to engage with the userbase or respond to queries. All of this will eat up valuable time.

### **4.2. Environmental Impact**

As a solely software project, Redreaper has a fairly limited environmental impact. The extent of it would probably be increased power usage by the server hosting the website, or perhaps by the computers of the users themselves. It's not clear that deployment of the site would even cause an increase in power consumption at all – perhaps the power would be used for other tasks instead by the users, or perhaps the server farm has excess capacity currently being wasted.

### **4.3. Social Impact**

This is one of the main goals of the project. As discussed earlier, by showing detailed analysis of Reddit users and subreddits built from publicly accessible data, Redreaper should demonstrate to its users just how much can be learned about a person or a community through software analysis. Coupled with the knowledge that this sort of analysis is being done privately, opaquely, by every major social media

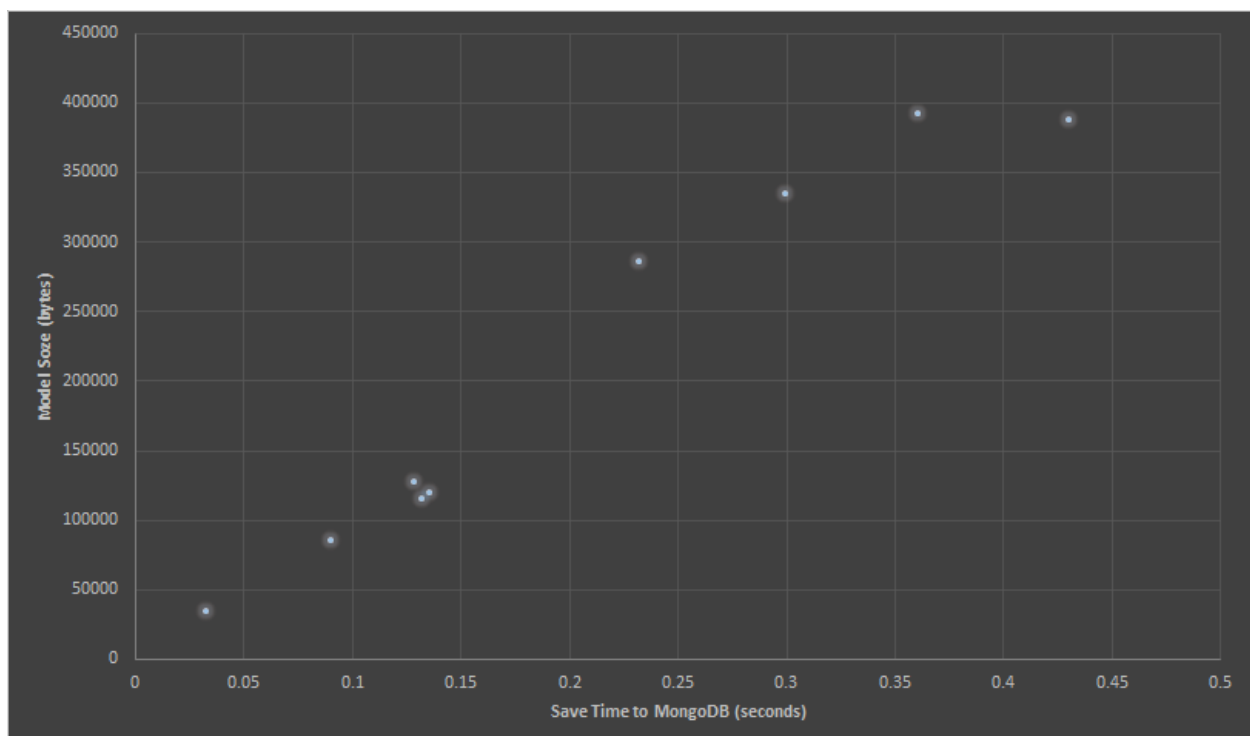
company there is, Redreaper should help spark insight into the new world of Big Data and how it affects them personally.

One notable example in our development came about from the “sentiment analysis” portion of Redreaper’s NLP output. For testing purposes, one team member input their own Reddit username, and found that Redreaper turned up with a fairly negative comment in their history. Finding that they weren’t comfortable with this in their history, they deleted the comment. This sort of response is likely to not be unique.

#### 4.4. Scalability

While Redreaper has not seen wide deployment, it has been built with that in mind. Part of this is the original choice to use the MEAN stack, which is widely tested and used for many websites. The website itself is very lightweight and loads very quickly, and the built-in framework of Bootstrap[15] for the overall website combined with the dynamic functionality of D3.js[16] for displaying results means that the website displays well across a wide variety of browsers and devices.

More importantly, since this is a project which would have to store lots of data on lots of users and subreddits, scalability means that our system must be able to handle a variety of database entries. We tested a set of different users, from ones with just a few posts to those with the maximum number of posts available from the Reddit API, as seen below:



As is reasonable, there is a linear relationship between user model size and database time. The database time is reasonable for the very largest of users, so a widely deployed Redreaper will be able to handle even extreme cases.

As for storage space, that broad sampling of users had a relatively even size distribution, and an average size of 209614 bytes. Assuming this is the average, and assuming the website ran with a simple 30GB SSD harddrive[14], then Redreaper could handle storing some 140 million estimated users – more than enough.

## 5. Conclusions

Redreaper has been a great project to work on. Starting out, we knew we wanted to create a product that would effectively perform interesting and varied social media analysis, and looking back we feel we have succeeded. During the public presentation of our project, each of the judges seemed interested in our software's capabilities, and seemed satisfied with our reasoning for the project's creation – both to present interesting analysis for Reddit users and subreddits, and also to show the kinds of things that modern software is capable of when given the vast amount of data available on us all. Even more satisfying was interest from fellow students, who came up to us and ran our program on their own Reddit accounts in fascination. We ourselves have found great value in analyzing our own accounts, and it says a lot about our overall social goal of promoting understanding of how much can be revealed by this kind of analysis that most of our team did not once share our personal Reddit account names with each other, lest they be analyzed publicly. Similar reticence was on display during our presentation day – some students seemed very interested, but were unwilling to have their data be shown in public. This indicates that a wide deployment, where users of Redreaper could look at the results in private, could very well be met with a very positive response.

There were also lessons learned about general software development, of course. These included the difficulties and benefits of working together on a software development team, the weighing that has to be done when deciding what exciting features will and won't make the cut for the final product, and the value of all the extra work that goes into planning, documentation, and gathering feedback throughout the development process. We certainly learned a lot, and expect to make use of that knowledge going forward, whether with further development of Redreaper or in our professional lives.

## 6. REFERENCES

- [1] "Reddit." *Wikipedia*. Wikimedia Foundation, n.d. Web. 15 Apr. 2016.  
<<https://en.wikipedia.org/wiki/Reddit>>.
- [2] "How The Internet\* Talks." *FiveThirtyEight*. N.p., 18 Nov. 2015. Web. 15 Apr. 2016.  
<<http://projects.fivethirtyeight.com/reddit-ngram>>.
- [3] "Karma Decay." *Karma Decay*. N.p., n.d. Web. 15 Apr. 2016. <<http://karmadecay.com/>>.
- [4] "Top Comment Karma." *Top Reddit Karma Users*. N.p., n.d. Web. 15 Apr. 2016.  
<<http://www.karmawhores.net/>>.
- [5] "Download Reddit Comment History." *Graph Reddit Comments*. N.p., n.d. Web. 16 Apr. 2016.  
<<http://www.roadtolarissa.com/javascript/reddit-comment-visualizer/>>.
- [6] "Redective." - *The Reddit Search Detective*. N.p., n.d. Web. 16 Apr. 2016.  
<<http://www.reductive.com/>>.
- [7] "SnoopSnoo - Reddit User and Subreddit Analytics." *SnoopSnoo*. N.p., n.d. Web. 16 Apr. 2016.  
<<http://snoopsnoo.com/>>.
- [8] "60 Amazing Reddit Statistics." *DMR*. N.p., 26 Feb. 2014. Web. 18 Apr. 2016.  
<<http://expandedramblings.com/index.php/reddit-stats/>>.
- [9] "MEAN.IO - MongoDB, Express, Angularjs Node.js Powered Fullstack Web Framework - MEAN.IO - MongoDB, Express, Angularjs Node.js Powered Fullstack Web Framework." *MEAN.IO*. N.p., n.d. Web. 19 Apr. 2016. <<http://mean.io/#!/>>.
- [10] "Vanak/red-reap." *GitHub*. N.p., n.d. Web. 19 Apr. 2016. <<https://github.com/vanak/red-reap>>.
- [11] "Software." - *The Stanford Natural Language Processing Group*. N.p., n.d. Web. 19 Apr. 2016.  
<<http://nlp.stanford.edu/software/>>.
- [12] "Reddit/reddit." *GitHub*. N.p., n.d. Web. 19 Apr. 2016.  
<<https://github.com/reddit/reddit/wiki/OAuth2>>.
- [13] "Stanford CoreNLP." – *a Suite of Core NLP Tools*. N.p., n.d. Web. 19 Apr. 2016.  
<<https://stanfordnlp.github.io/CoreNLP/>>.
- [14] "Best Web Hosting Features, Vps Hosting, Dedicated Hosting, by DreamHost – DreamHost." *Best Web Hosting Features, Vps Hosting, Dedicated Hosting, by DreamHost – DreamHost*. N.p., n.d. Web. 19 Apr. 2016. <<https://www.dreamhost.com/hosting/>>.



[15] "Bootstrap · The World's Most Popular Mobile-first and Responsive Front-end Framework." *Bootstrap · The World's Most Popular Mobile-first and Responsive Front-end Framework*. N.p., n.d. Web. 02 May 2016. <<https://getbootstrap.com/>>.

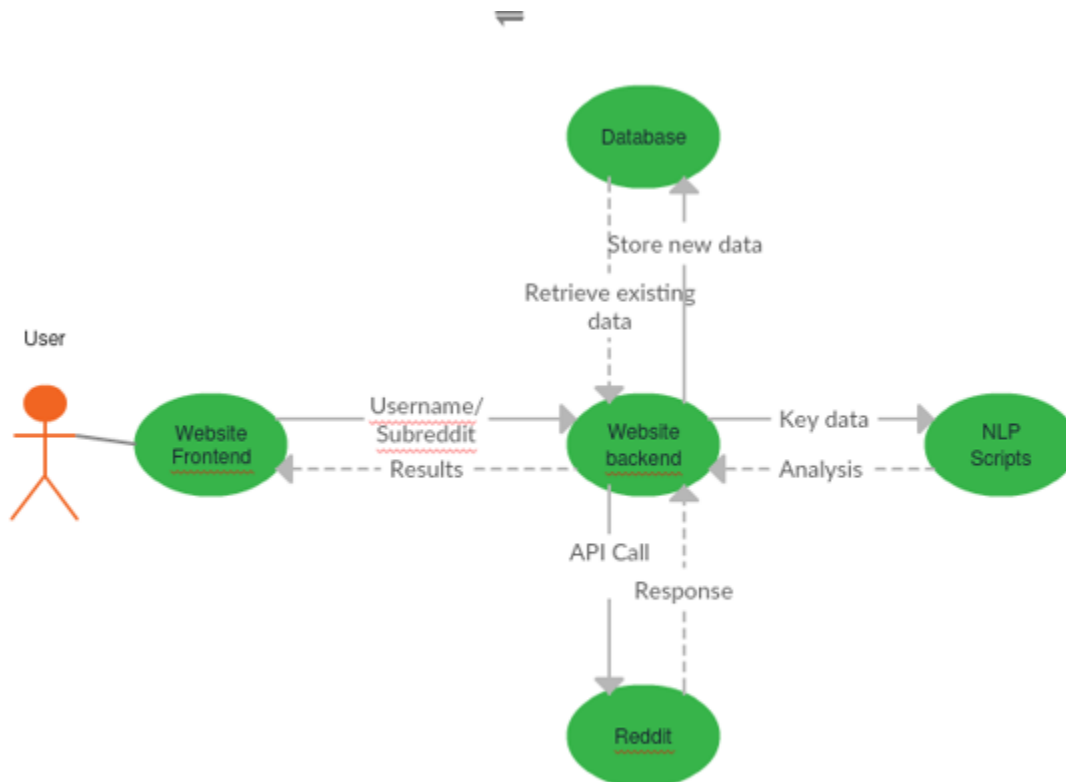
[16] "D3.js - Data-Driven Documents." *D3.js - Data-Driven Documents*. N.p., n.d. Web. 02 May 2016. <<https://d3js.org/>>.

## 7. Appendix

The following sections are various relevant documents produced throughout the Redreaper project development. These can also be found on Redreaper's GitHub[10].

### 7.1. Use Case Discussion

The following comes from a document explaining use cases made mid-way through development.



Reddit Reaper has two core functionalities, both of which operate on the same basic model. A user interacts with Reddit Reaper by loading the website, where they are given the option to submit either a subreddit or a username for analysis. Reddit Reaper then makes an API call if data is needed, then analyzes that data using NLP scripts, statistics, and mathematics, then stores the data. Finally, the output is presented to the user on the website. If the data required was already stored, from either a previous identical request or a database population effort, then the website backend simply retrieves it from the database and presents it to the user. While the user may have any number of reasons for requesting the specific analysis that they do, delivering that analysis is the one and only function of this project.

While users may in fact be interested in only certain pieces of analysis, the time it would take to select specific desired outputs could easily eclipse the extra processing time as well as needlessly complicate matters. Moreover, the goal of Reddit Reaper is to provide unexpected insight, which will only be more difficult if users limit the information provided to them. Thus, the program has only two straightforward uses.

#### Basic Use Case 1:

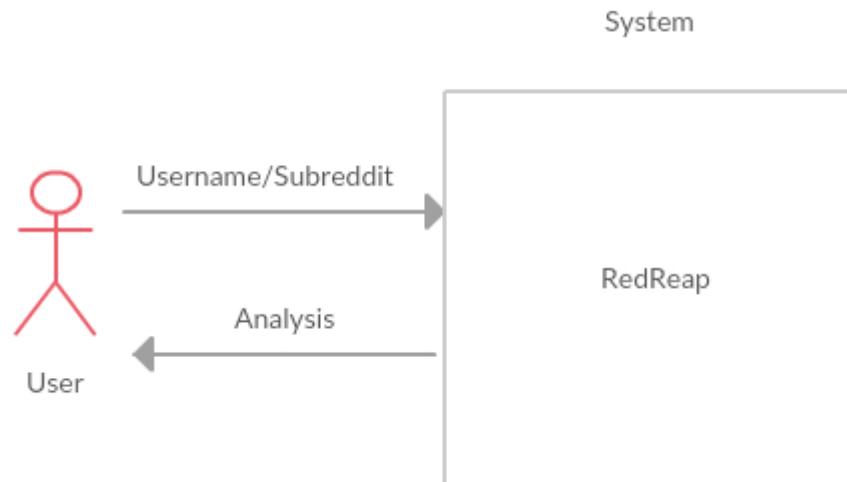
"I want to see analysis on my Reddit account."

1. Access RedReap website
2. Enter Reddit account username for analysis
  - a. Goes into a labelled text field, press enter or click button to start
3. Wait for data to be retrieved/analysis to be performed
  - a. Loading icon/information will be displayed
4. Website displays data and analysis on requested user
  - a. This information is generally organized by category
  - b. Sidebar links on the display page allow quick access to categories
  - c. Some graphs and tools can be interacted with by the user
5. Use top bar of website to return to main page for a new analysis

#### Basic Use Case 2:

"I want to see analysis on a subreddit."

6. Access RedReap website
7. Enter subreddit name for analysis
  - a. Goes into a labelled text field, press enter or click button to start
8. Wait for data to be retrieved/analysis to be performed
  - a. Loading icon/information will be displayed
9. Website displays data and analysis on requested subreddit
  - a. This information is generally organized by category
  - b. Sidebar links on the display page allow quick access to categories
  - c. Some graphs and tools can be interacted with by the user
10. Use top bar of website to return to main page for a new analysis



#### Detailed Use Case 1:

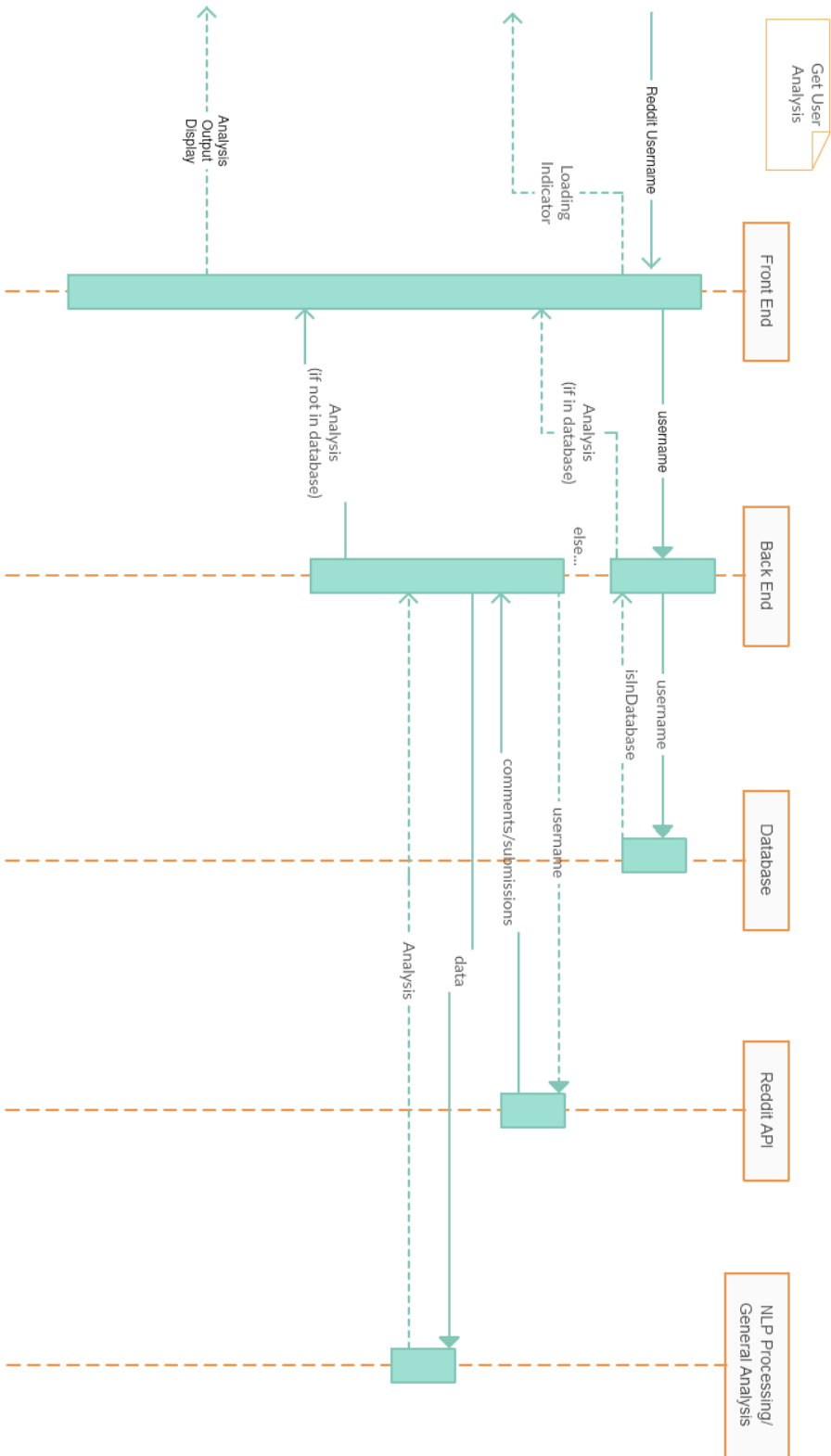
1. Access Reddit Reaper website
2. Enter Reddit account username
  - a. Username is sent to backend
    - i. Check if user has already been analyzed recently
  - b. If not already analyzed:
    - i. Make API call for user submissions and comments
    - ii. Compile basic statistics and metadata
    - iii. Analyze compiled data
    - iv. Send comments/submissions to NLP scripts for analysis
    - v. Store all compiled data and analysis into database entry for user
    - vi. Send data to frontend
  - c. If already analyzed:
    - i. Access data and analysis on user from database
    - ii. Send data to frontend
3. Website displays data and analysis on requested Reddit account
  - a. While waiting for information from backend, display loading indication and information
    - i. Loading icon, time estimate if possible
  - b. Once data is available, load analysis display page
    - i. Also load a sidebar that can be used to jump between sections
  - c. Can return to main page and thus return to Step 1

#### Detailed Use Case 2:

1. Access Reddit Reaper website
  - a. Click on top bar option for "Subreddit Analysis"
2. Enter subreddit name
  - d. Subreddit name is sent to backend
    - i. Check if subreddit has already been analyzed recently
  - e. If not already analyzed:
    - i. Make API call for subreddit submissions and comments
    - ii. Compile basic statistics and metadata
    - iii. Analyze compiled data
    - iv. Send key data to NLP scripts for analysis
    - v. Store all compiled data and analysis into database entry for subreddit
    - vi. Send data to frontend
  - f. If already analyzed:
    - i. Access data and analysis on subreddit from database
    - ii. Send data to frontend
4. Website displays data and analysis on requested Reddit account
  - a. While waiting for information from backend, display loading indication and information
    - i. Loading icon, time estimate if possible
  - b. Once data is available, load analysis display page
    - i. Also load a sidebar that can be used to jump between sections
  - c. Can return to main page and thus return to Step 1
    - i. Can also go directly back to subreddit analysis page

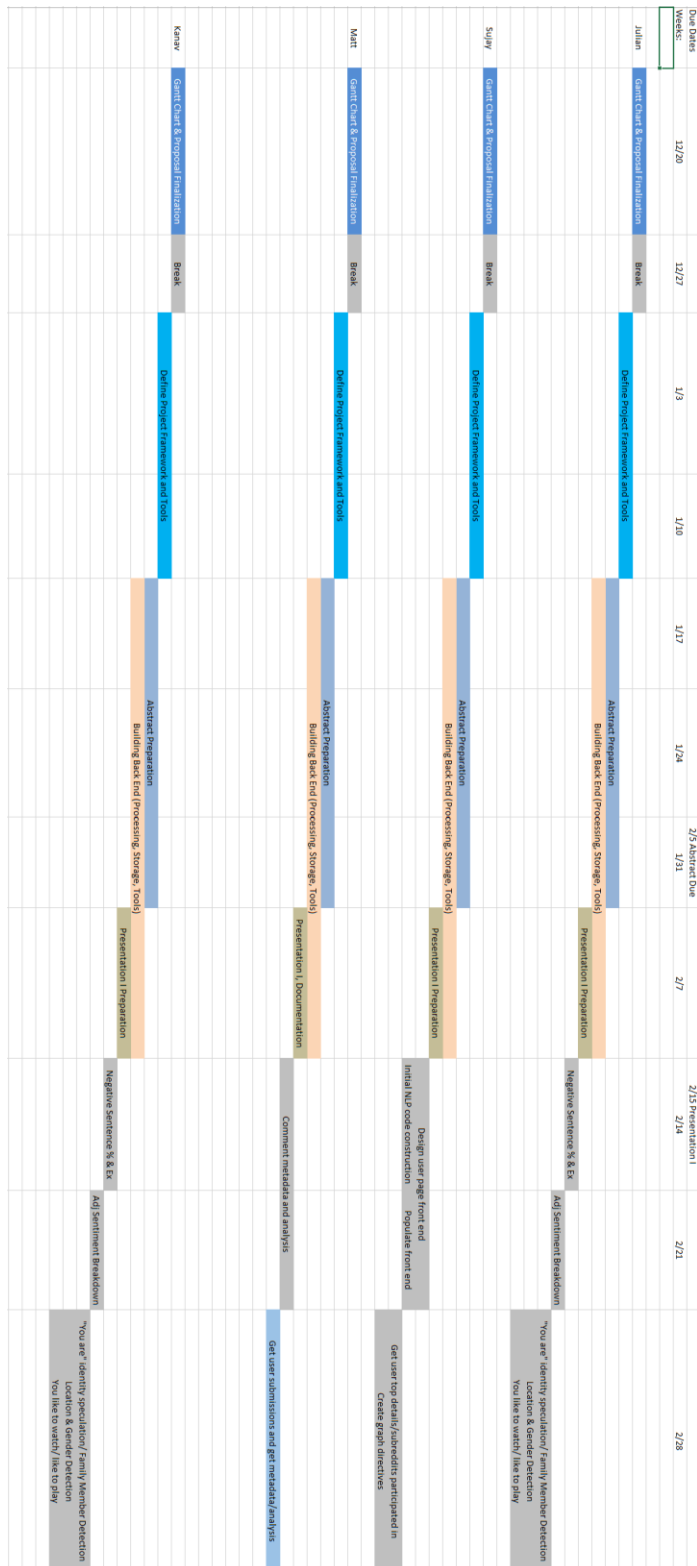
#### Possible variations on use cases:

1. Requested user or subreddit does not exist
  - a. Display error, return to main page
2. User wants most recent possible analysis
  - a. After website displays requested analysis retrieved from database, display option to make a new API call and perform new analysis
3. User accidentally requests analysis on wrong subject, or otherwise changes their mind
  - a. Display button to cancel ongoing analysis
4. User or subreddit exists, but has limited data e.g. no comments
  - a. Display notice along with affected analyses
5. Website is out of API requests temporarily
  - a. Return error
  - b. Give option to wait or to authenticate through Reddit
6. Website is under heavy load and can't process data
  - a. Return error, do not attempt to process new requests
7. Portion of analysis fails
  - a. Catch error internally
  - b. Continue on to produce and display other analyses
  - c. Display error for affected section



## 7.2. Gantt Chart

The following Gantt chart was adjusted in order to better reflect duties mid-semester.



|  |  |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |
|--|--|------|-----------------|------|------|-----|--------------------------------------|------|-------------------|------|------|-----------------------------|-----|
| 3/6  | 3/18 Intern Report   | 3/13 | Presentation II | 3/20 | 3/27 | 4/3 | 4/15 Final Report Draft/Poster Draft | 4/10 | 4/22 Final Poster | 4/17 | 4/24 | 5/1 Final Report/60 Sec Vid | 5/1 |
| Refinement of You Are & Family<br>You like to discuss/You like<br>Sentiment over time/By Subreddit           | N/A accuracy improvement using grammar check & spelling statistics |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |
|  | Subreddit Equivalents of all features & trending topics within sub |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |
|  | 4/15 Final Report Draft/Poster Draft                               |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |
| Design subreddit page front end/Get subreddit data and analysis<br>Work on intern report and Presentation II | Implement subreddit features in front end                          |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |
|  | Test/Improve existing code, add existing features if time allows   |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |
|  | 4/15 Final Report Draft/Poster Draft                               |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |
| Get subreddit data and get metadata/analysis<br>Work on intern report and Presentation II                    | Test/Improve existing code, add existing features if time allows   |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |
|  | 4/15 Final Report Draft/Poster Draft                               |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |
|  | 4/22 Final Poster  |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |
| Refinement of You Are & Family<br>You like to discuss/You like<br>Sentiment over time/By Subreddit           | Subreddit Equivalents of all features & trending topics within sub |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |
|  | 4/15 Final Report Draft/Poster Draft                               |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |
|  | 4/22 Final Poster  |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |
| Refinement of You Are & Family<br>You like to discuss/You like<br>Sentiment over time/By Subreddit           | N/A accuracy improvement using grammar check & spelling statistics |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |
|  | Subreddit Equivalents of all features & trending topics within sub |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |
|  | 4/15 Final Report Draft/Poster Draft                               |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |
| Refinement of You Are & Family<br>You like to discuss/You like<br>Sentiment over time/By Subreddit           | Subreddit Equivalents of all features & trending topics within sub |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |
|  | 4/15 Final Report Draft/Poster Draft                               |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |
|  | 4/22 Final Poster  |      |                 |      |      |     |                                      |      |                   |      |      |                             |     |