*Abstract*—In this paper, we will outline the process of data fetching using GCP and Spark, data storage with a VM instance, and data analysis with Dataprep and Google BigQuery. To understand the latest Reddit trends, we will calculate the frequency of topics within the header of a post. The purpose of this paper is to understand how API’s work with GCP and Spark to fetch Reddit data relevant to recent trends.

*Reddit – a social news website and forum where content is socially curated and promoted by site members through voting.*

*Subreddit - a page pertaining to a particular subject/community on Reddit. Created and managed by that community.*

*GCP (Google Cloud Platform) – a suite of cloud computing services that runs on the same infrastructure that Google uses internally for its end-user products.*

*Dataprep (by Trifacta) – an intelligent data service for visually exploring, cleaning, and preparing structured and unstructured data for analysis, reporting, and machine learning.*

*Google BigQuery – a fully managed enterprise data warehouse that helps you manage and analyze your data with built-in features like machine learning, geospatial analysis, and business intelligence.*

*SQL (Structured Query Language) - a standardized programming language used to manage relational databases and perform operations on the data in them.*

*API (Application Programming Interface) – a software intermediary that allows two applications to talk to each other.*

# **I.Introduction**

This paper is a final project report for our Cloud Computing class, at the start of the summer we were all broken up into groups which we would work with on a couple of assignments and eventually this final project. When it came time to decide on a topic for our proposal, we were all flirting with doing something with Google Cloud since it was a system no one had worked with before therefore yielding the most opportunities for us to learn something new. Python was then the natural choice for a programming language since we would be fetching and analyzing a significant amount of reddit comment data from a very controversial time in recent history - March 2020. This was the month COVID-19 really started to hit the United States hard so we thought it would be interesting to see what the reddit data would be during that month. We intend to weed out the harshest and kindest subreddits during this time to better understand what worried people and what people used as a distraction.

# **II. Creating a VM Instance in Google Cloud**

## **VM Configuration for Free**

Google Cloud offers a free $300 of credit to use their resources up to 90 days after you have created your account. So, we had to be smart about selecting the right specs for the hardware we would need to process approximately 200 GB of reddit data. We got to select the closest regional server to our location for the best network performance possible. For the machine type we ended up going with the e2-standard-2 which was 2 cores and 8 GB of RAM memory. This ended up being perfect for us since it has a $55.09 monthly service charge estimate and it only took a little over an hour to process our data. Finally, we went with a disk size of 350 GB since it hardly cost that much extra and we would only be using it for a couple of days.

## **GCS Bucket**

Within GCP, Google offers cloud storage (GCS) as well. Dubbing their storage devices “buckets”. These buckets act as a staging area to hold the data that we’ll analyze with Google BigQuery. We first downloaded our dataset onto the instance using “wget”, but after that we had to copy said data into our storage bucket using this command:

# 

# **III. Dataprep & BigQuery**

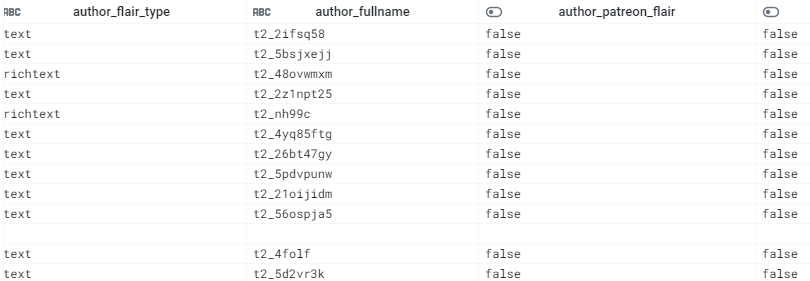
## **Dataprep RR (Review & Recipes)**

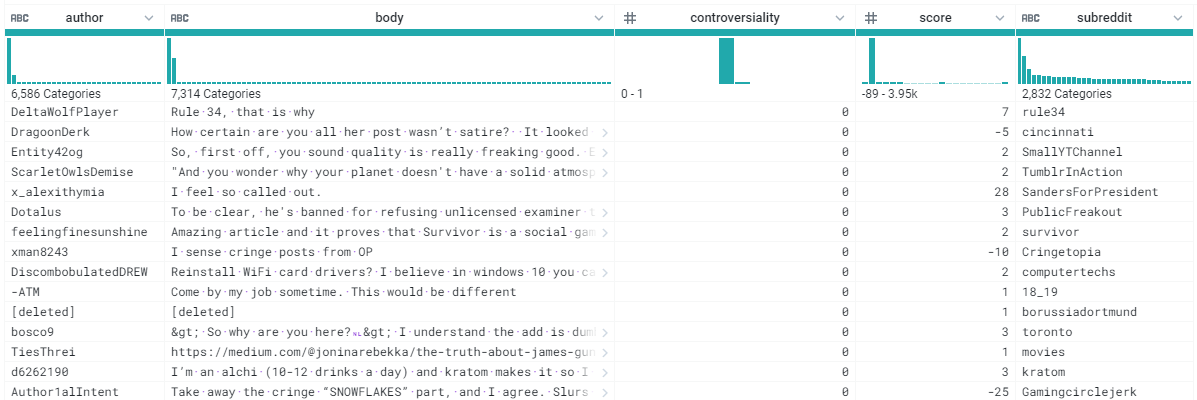
This being our first time using Dataprep we had quite a bit to figure out on our own, mostly through trial and error. We had to reprocess the data multiple times to get it right, to no fault of Dataprep it was definitely a user-error. After we got past the growing pains of rookies to Dataprep, we absolutely loved it. The UI struck the perfect balance of simplicity and functionality. It takes something as complex as data manipulation and transformation, into a joyful and informative experience where anything you want to know or do to your data is at your fingertips.

One way it made the transformation of our reddit data easy, was with “recipes”. Recipes essentially offer a way to look at any data you have in a GCS bucket and filter or clean it into whatever you want. Have some values in a column that didn’t come out right? No problem, click on the column you want to clean and it’ll take out all instances of any value you need. Want to take out a couple columns you won’t need to make that date more seamless? Done. This is what we mainly used Dataprep for, to show the harshest and kindest subreddits during the month and year of March 2020 we only needed 5 columns out of the original 42. Which were:

* **Author** (string) - the author of the comment that was written.
* **Body** (free text) - the actual contents of the comment.
* **Subreddit** (string) - name of the subreddit where the comment posted.
* **Score** (integer) - the number of people who upvoted the comment.
* **Controversiality** (0 or 1) - the state of whether the comment was controversial or not.

Next, here is a view of how the data looked before and after the Recipe was applied:

**Before:**

**After:**

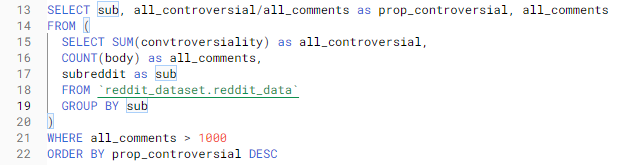
As you can see, the data before had a lot of unnecessary fields for what we are trying to interpret from the data.

## 

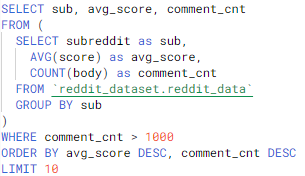
## 

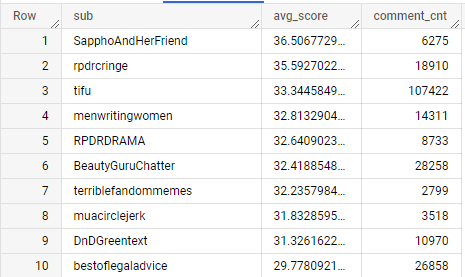
## 

## **BigQuery**

The way I like to think of Google BigQuery as a glorified SQL Workbench. It does exactly what you need it to do, you can open up any dataset that you’ve transformed using Dataprep or directly from GCS which is very convenient. It’s also easy to run SQL commands on said data, it gives you a nice console to run things on and once you run a command it returns the data based on the SQL command in a great format.

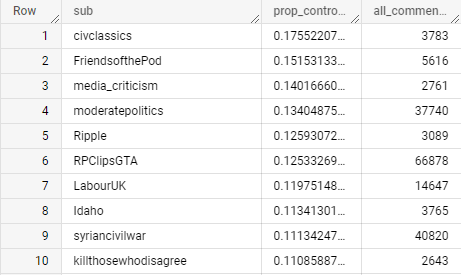
Here is an example of a SQL command I ran to find the top 5 subreddits with the highest average score (Kindest Subreddits):



Output:

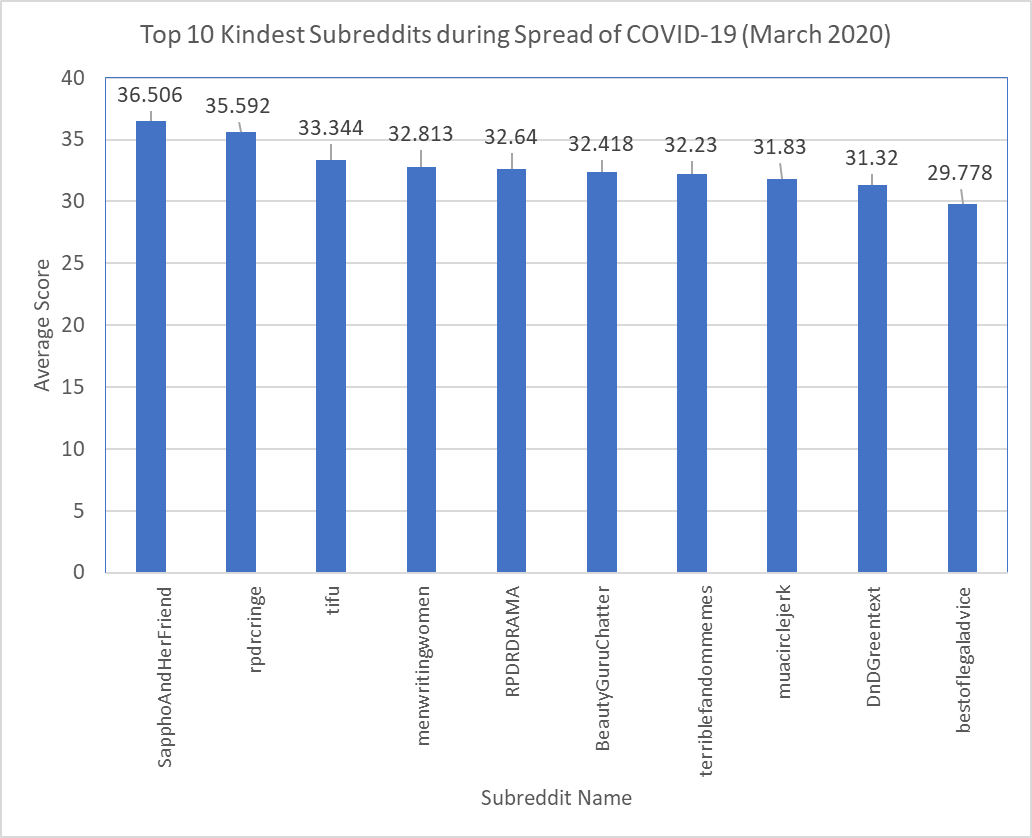
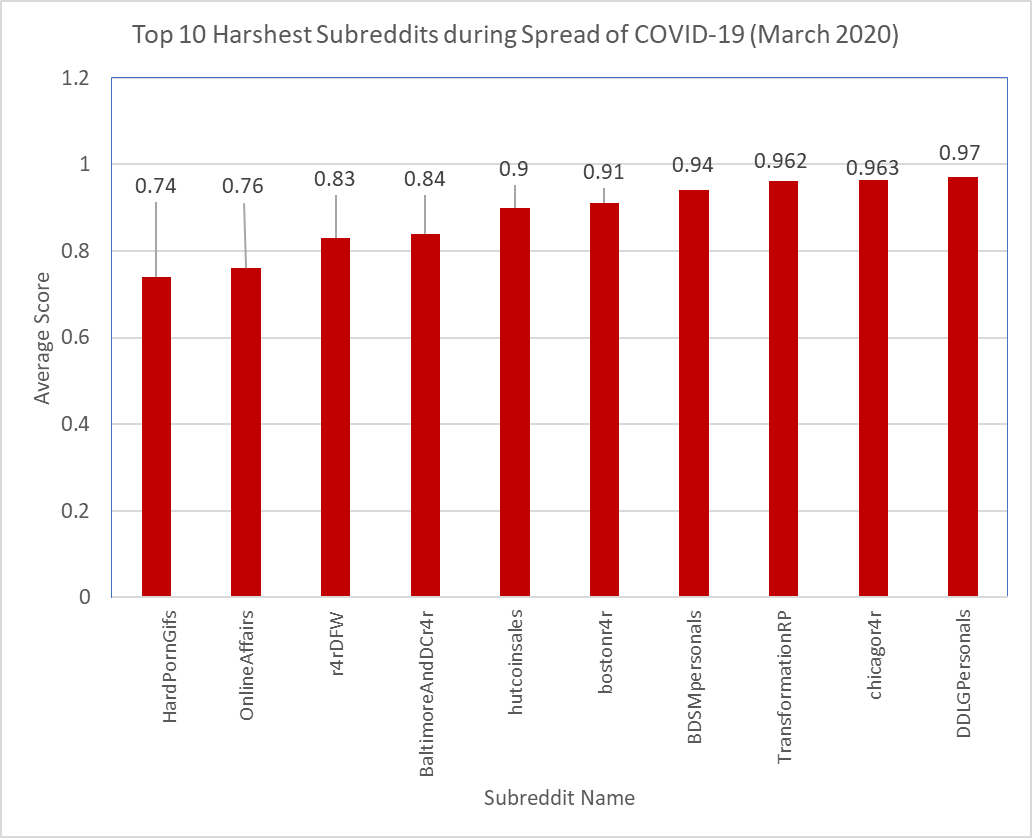
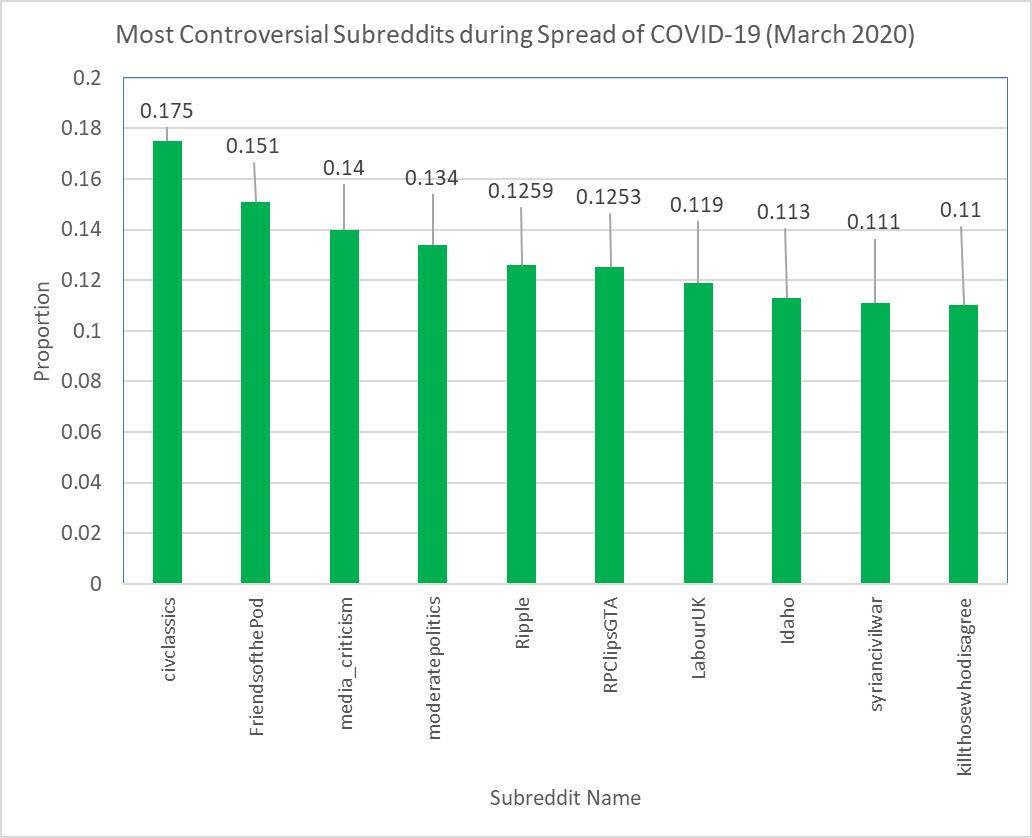
Example 2: This next query finds the most controversial subreddits based on proportion of comments:

Output:

**IV. Results**

1. **Summary**

All in all this project was an absolute blast to work on. We all learned so much from the Google Cloud Platform, from creating VM’s with the proper hardware to support our operations to transforming and working on data in Dataprep and BigQuery. It proved to be extremely fascinating to take a look into the past (March 2020 to be specific) during such a pivotal point in human history like the spread of COVID-19 to see what people were talking about on a huge social news platform like Reddit. It’s a real eye-opener to compare that to our own experiences during that time and how we all handled it.

1. **Kindest Subreddits**
2. **Harshest Subreddits**
3. **Most Controversial Subreddits**

# **V. References**

1. <http://www.nicksun.fun/ds/2020/03/09/analyzing-reddit-comments.html>
2. <https://www.trifacta.com/resource-library/leveraging-cloud-dataprep-by-trifacta-bigquery-and-looker-for-marketing-analyticsleveraging-cloud-dataprep-by-trifacta-bigquery-and-looker-for-marketing-analytics/>
3. <https://cloud.google.com/bigquery/docs/datasets>
4. <https://rpubs.com/SmilodonCub/586863>
5. <https://stackoverflow.com/>