**SI 618 Fall 2019 Homework 6 (100 points)**

On the Hadoop cluster, I have put the following file in HDFS:

hdfs:///var/umsi618/hw6/review.json

This file was downloaded from <http://www.yelp.com/dataset_challenge> (you cannot share the content with others without going through the approval procedure yourself). The format of the data is explained in the “Notes on the Dataset” section at <https://www.yelp.com/dataset/documentation/main>.

**Finding Useful & Useless Reviews**

The goal of this question is to perform a similar analysis from lecture, but instead of looking at the reviews rating, we’ll look at how useful a review is. We can imagine that there might be trends between useful and useless reviews that Yelp users wrote. Do you think that’s true? Let’s check!

To answer this question, you are going to use Spark to compute the log-likelihood ratio. for each word in a useful and useless review. Use *yelp\_sentiment.py* as a starting point, you will need to edit it to look at the **useful** field, instead of stars.

Here is an example of one useful review, we can see here that 6 users have marked it as being useful. That’s pretty good!

|  |
| --- |
| {'business\_id': 'ujmEBvifdJM6h6RLv4wQIg',  'cool': 0,  'date': '2013-05-07 04:34:36',  'funny': 1,  'review\_id': 'Q1sbwvVQXV2734tPgoKj4Q',  'stars': 1.0,  'text': 'Total bill for this horrible service? Over $8Gs. These crooks actually had the nerve to charge us $69 for 3 pills. I checked online the pills can be had for 19 cents EACH! Avoid Hospital ERs at all costs.',  **'useful': 6,**  'user\_id': 'hG7b0MtEbXx5QzbzE6C\_VA'} |

We define a **useful** review as any review that has a useful rating greater than 5. A **useless** review is any review that has a useful rating equal to zero. The output should be a text file of words and their log-likelihood ratio sorted in descending order. You do *not* *need* to convert each word to lowercase. Your results should look like:

* si618\_hw6\_desired\_output\_usefulreview.txt
* si618\_hw6\_desired\_output\_uselessreview.txt

**What to submit:**

Your Spark code should run as a standalone application on the Cavium cluster.

* Submit a **zip file** named si618\_hw6\_youruniqname.zip containing:
  + si618\_hw6 \_youruniquename.py
  + si618\_hw6 \_youruniquename\_usefulreview.csv
  + si618\_hw6 \_youruniquename\_uselessreview.csv