EECS 4415 - Big Data - Project Proposal
Computing Representative Sets of Terms for Subreddits

Ibrahim Suedan
York University
Toronto, Canada
isuedan@my.yorku.ca

Ken Tjhia
York University
Toronto, Canada
hexken@my.yorku.ca

Qijin Xu
York University
Toronto, Canada
jackxu@my.yorku.ca

ABSTRACT

In this project we propose a big data architecture to determine a set of the most relevant terms for each of the n most commented in subreddits. To achieve this, we first determine the n most commented in subreddits, then we consider each of these as a document and compute the term frequency - inverse document frequency (TF-IDF) scores for each word in each document. The k words of each subreddit with the highest TF-IDF scores then form its representative set of terms.

KEYWORDS

MapReduce, BigQuery, datasets, Reddit, TF-IDF, information-retrieval

1 INTRODUCTION

Reddit is one of the most visited websites as of 2019 [3], and marketing firms hoping to take full advantage of this require insights into the trends and culture of individual subreddits. Term frequency - inverse document frequency (TF-IDF) is a popular term-relevancy metric in the information retrieval community [2], and has previously been used to construct numerical vector representations of documents, which is especially useful for machine learning tasks. Here we propose a big data architecture to compute the TF-IDF scores of words in subreddits, which we hope will allow one to gain insights into the subreddits culture. Specifically, we will first identify the set P of the n most popular subreddits, ranked by their number of comments, then for each $p \in P$ we compute the TF-IDF scores for every word in p (the set of comments which comprise p). Then, for each subreddit $p \in P$ we record the k words with the highest TF-IDF scores as the most relevant terms for p; these words form our representative set of terms for p.

Advertisers can then use this information to select which subreddits to focus their efforts on (for example target subreddits with relevant terms that are aligned with the advertising goals), to tailor campaigns to particular subreddits (for example use subreddit relevant terms in the advertising text), and to evaluate the performance of ongoing campaigns (for example by monitoring the changes in the representative sets). More generally, the TF-IDF scores can be used as subreddit features for machine learning tasks such as subreddit classification or recommending subreddits to users. Hence, the results of our analysis can potentially provide valuable information for businesses such as marketing or public relations firms, even Reddit itself, and will provide an open dataset of subreddit representations for academics and other interested parties.

```
{comment : {
    {
        "author": "ExampleCommenter",
        "author_flair_css_class": null,
        "author_flair_text": null,
        "body": "This is an example comment",
        "created_utc": 1270637661,
        "id": "c0nn9iq",
        "link_id": "t3_bne3u",
        "parent_id": "t1_c0nn5ux",
        "score": 2,
        "subreddit": "askscience",
        "subreddit_id": "t5_2qm4e"
    }
}}
```

Figure 1: Reddit comments JSON schema

2 DATA DOMAIN

The Reddit website is organized into different subreddits, where each subreddit contains a number of posts, and each post contains a number of comments. We will begin with a historical dataset [1] of all comments posted to Reddit during a particular month. This dataset is a text file containing an array of JSON objects following the schema of Figure 1, where the relevant fields for our task are the *subreddit_id* and *body* fields. Depending on how long processing for the first month takes, we may extend the analysis to more historical months. For the current (on-going) month, we will use the Reddit API to get all comments up to that point (following the same schema of Figure 1), then periodically retrieve newly posted comments and update the TF-IDF scores.

3 ARCHITECTURE

3.1 Batch

Our batch analysis ingests data from PushShift [1]; we download the datasets corresponding to the months of interest and store them in a Hadoop distributed file system (HDFS). We will then use MapReduce jobs for preliminary cleaning and to compute the TF-IDF scores, then load the scores into a BigQuery database connected to a Tableau dashboard, where we will display various subreddit level statistics such as the total number of comments and the words with the k greatest TF-IDF scores.

3.2 Streaming

Our streaming analysis ingests data (on a single computer) via queries to the Reddit API, retrieving all comments posted this month

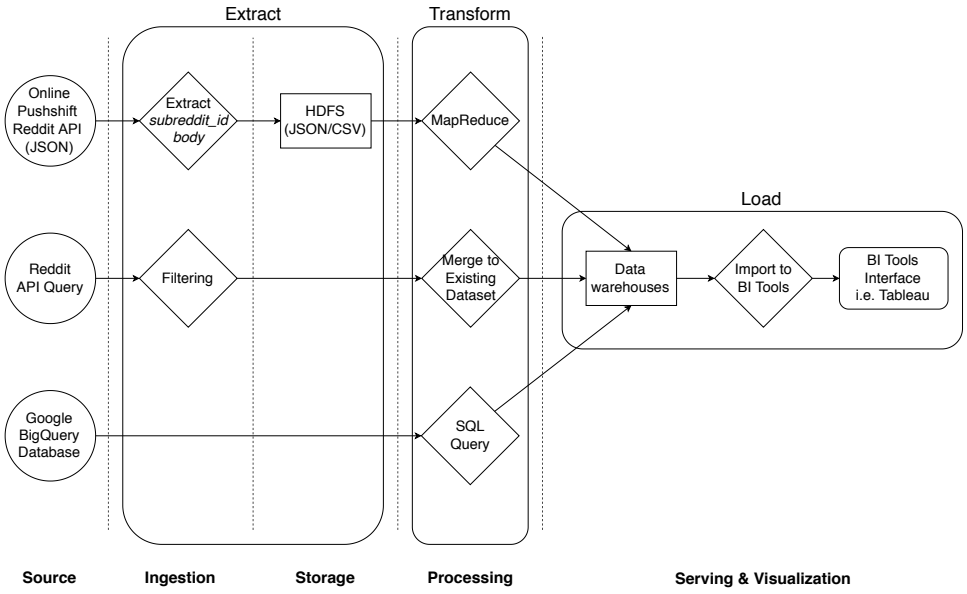

Figure 2: Proposed Architecture (with optional BigQuery data source)

up to the present time. We will process the comments as we retrieve them and keep only the minimum data necessary for computing and subsequently updating the TF-IDF scores. This data will be stored in another database (to be determined) that is also connected to the Tableau dashboard. We will periodically (period to be determined) retrieve all newly posted comments and update the database.

3.3 Expected Limitations and Difficulties

One problem we anticipate is that while we would like to work on the most recently posted monthly batch dataset, the file is quite large at ~12GB and we anticipate that the MapReduce (on a single computer cluster..) jobs will take some time to complete.

We will use the same vocabulary of words across all subreddits when computing TF-IDF scores, so another issue is determining what this vocabulary should be. There may be words found only in one subreddit which are completely representative of that subreddit, and we would like to identify these words and include them in the vocabulary.

3.4 Scalability and Re-usability

We aim to build the separate components of our pipeline in a general way so that they can be re-used with similar data. For example, the TF-IDF computations will compute the TF-IDF scores of words given a particular vocabulary and an arbitrary set of documents; they will not depend on the schema presented in Figure 1. The BigQuery database is also agnostic of the Reddit comments schema; only the data ingestion and some of the elementary cleaning stages will be written specifically for the schema in Figure 1. Because only the data ingestion portions of our architecture depends on Reddit, we can easily adapt the system to perform the same analysis on any abstract set of documents.

Since our batch analysis already uses a distributed file system, the MapReduce programming model, and BigQuery, our architecture may be easily scaled horizontally. For the streaming portion, we intend to use a single computer; however, if required, we can adopt a distributed stream processing framework such as Apache Storm.

4 ANALYSIS AND EVALUATION

TF-IDF is already an established metric for creating numerical vector representations of documents. To evaluate how well our top k words (per subreddit) represent each subreddit, we will first assign each subreddit α a vector $v_\alpha \in R^{|V|}$, where $|V|$ is the size of our vocabulary, and the kth component of v_α is 1 if the word corresponding to this component is present in α's set of k words with greatest TF-IDF, and 0 otherwise. We then cluster these vectors and visualize in two-dimensions (for example using t-SNE or PCA) to observe how well the clusters correspond to our subjective assessments of subreddit similarities. We can also compare this clustering with one based on the whole TF-IDF vectors to determine whether our choice of k was adequate, and also look at the actual k words for each subreddit and subjectively determine whether they are a representative set.

To determine whether our streaming analysis produces useful results, we will monitor the changes in TF-IDF scores and observe whether they are significant for the chosen retrieval intervals.

REFERENCES

[1] PushShift. 2019. *PushShift Reddit API.* Retrieved Nov 11, 2019 from https://files.pushshift.io/reddit/comments/

[2] Shahzad Qaiser and Ramsha Ali. 2018. Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents. *International Journal of Computer Applications* 181 (07 2018). https://doi.org/10.5120/ijca2018917395

[3] SimilarWeb. 2019. *Combined desktop and mobile visits to Reddit.com From February 2019 to July 2019.* Retrieved Nov 11, 2019 from https://www.statista.com/statistics/443332/reddit-monthly-visitors/