Kenny Scharm and Jiashen Cao
CS 8803
April 28, 2019

Final Project: Analyzing Trending Twitter Topics via MapReduce & Key Phrase Extraction

Introduction

For our final project of the semester, we implemented a method to analyze real-time trending topics on Twitter using the MapReduce framework. Before feeding the data to our MapReduce framework, we first extract key phrases in tweets using the Microsoft Azure Cognitive Services API. This specific API takes in multiple text documents and extracts interesting words or phrases from the text. Once the tweets have been analyzed for key phrases, we input the data to our mapper function. This scraping process constantly pulls new data from Twitter and periodically sends this data to the master node within the MapReduce framework. The Twitter scraper sends data to the master node with a period that is proportional to the size of the data. We set a maximum size limit of 100Kb for the scraper. Each time the scraper reaches the size limit, it sends its current data to the master node and flush its local data. We chose this approach instead of constantly streaming tweets because MapReduce jobs take some time to complete (~30 seconds).

Map Function

The mapper function takes the preprocessed Twitter data and records the frequency of the unique words and phrases. Similar to the first MapReduce workshop in class, we take advantage of a Hadoop cluster running on an HDInsight instance. Currently, we are running the smallest, lowest-cost cluster for testing purposes. To increase the performance of the cluster, we would need to increase the compute power of the head nodes. This would help reduce a major bottleneck in our system--MapReduce job completion time.

Reduce Function

The reduce function takes the mapper data and aggregates it based on the key word. In addition to combining the intermediate data, the reducer nodes sort the results by decreasing frequency. Once the reduce phase has completed, the results are stored in Microsoft Azure Blob Storage.

Architecture

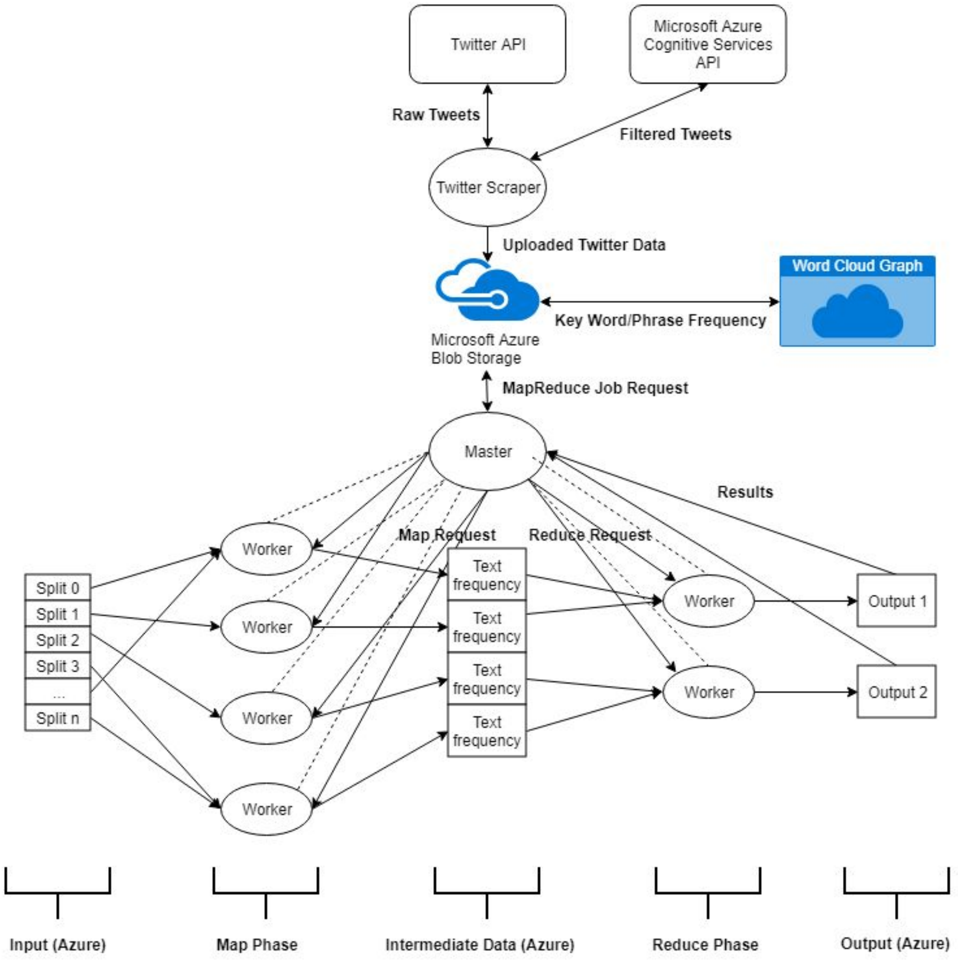

Figure 1. Proposed MapReduce architecture

Analysis & Results

We analyze the results of the output file stored in Microsoft Azure Blob Storage by generating a word cloud graph based on the top key words and phrases. The graph updates dynamically as the output file is updated in Azure.

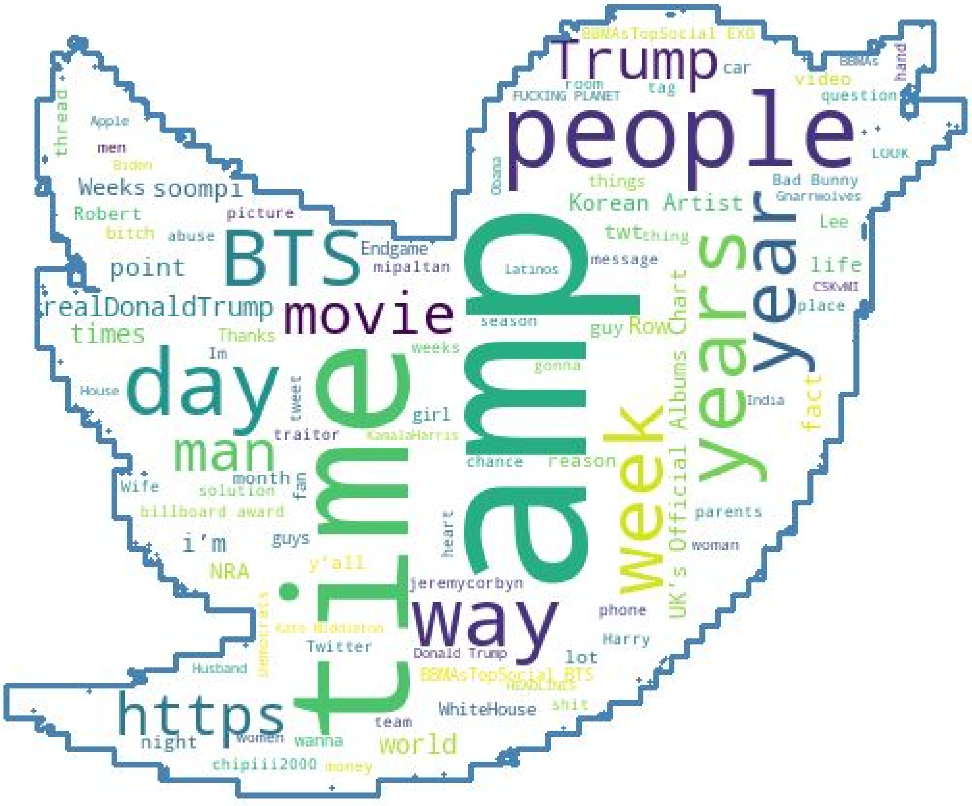

Figure 2. Generated word cloud graph based on key word/phrase frequencies in real-time tweets.

With our current parameter configuration, the graph updates approximately every 45 seconds to a minute. This time could be greatly reduced if we had access to more Twitter data, which is limited by the number of API calls you can make per second, or if upgraded our Hadoop cluster (more nodes, more compute power, etc.). It is important to note that some variance in the update time is expected because of variance in both scraping Twitter data and MapReduce jobs.