Dataset Ingest and Analysis Service

Aaron Gumabong, Undergraduate, CPSC

Michael Rodriguez, Undergraduate, CPSC

Saytu Singh, Undergraduate, CPSC

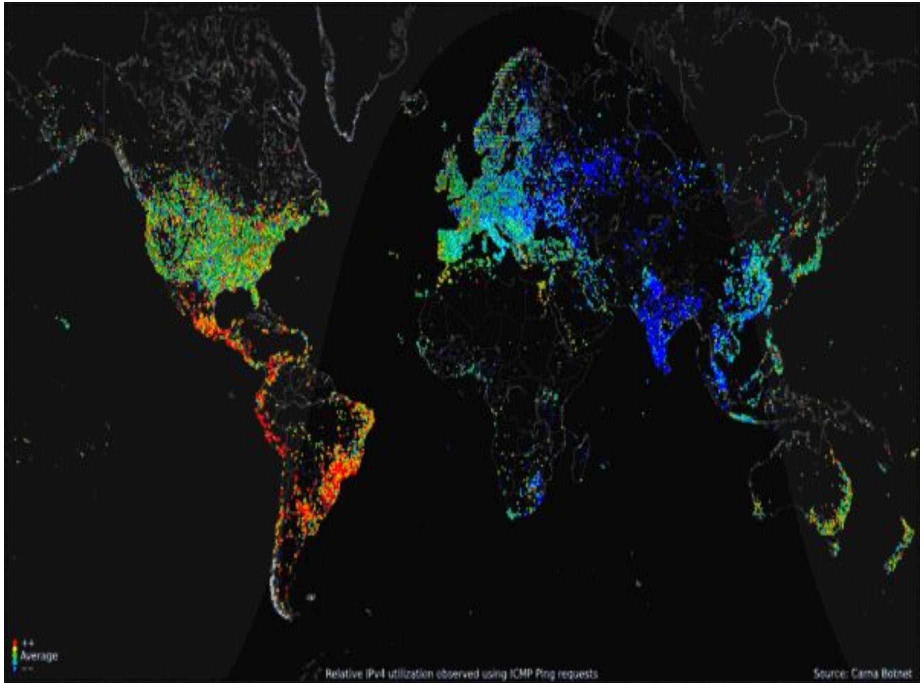
Hasib Ziai, Undergraduate, CPSC

Project Description

The main objective of our project is to create a cloud-based service that analyzes and provides a detailed result about the dataset.

Project Inspiration

In 2012, a security researcher discovered an ‘exploit’ which allowed the researcher to gain access to a victim’s network via Linksys routers and certain internet connected devices. They wrote a script which scanned all IPv4 addresses and patched the vulnerability, then scanned the internet for more devices to patch, effectively creating a botnet. The results of the scan was then sent to the researcher’s ‘command’ servers, resulting in the “Internet Census of 2012”. The map to the right shows a world map of 24-hour relative average utilization of IPv4 addresses observed using ICMP pings.



Relative IPv4 utilization observed using ICMP Ping requests

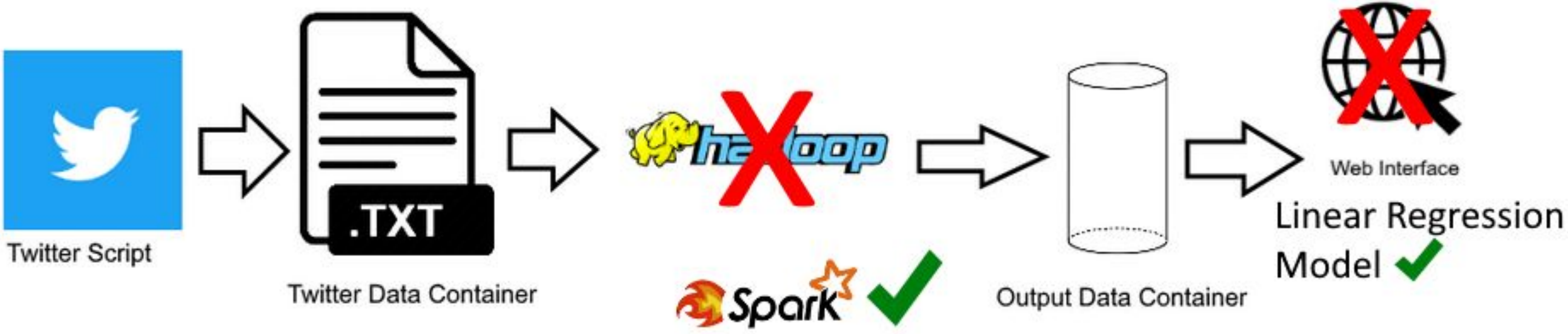
Source: Carna Botnet

Use Case

- The user’s goal will be to view detailed analytics about a dataset through a simple process.

- The user will specify the dataset to be analyzed beforehand, and the type of analysis for the dataset will be defined preemptively. They will then be able to view the selected information from a simple file.

Design


Twitter Script
.TXT
Twitter Data Container
hadoop
Spark
Output Data Container
Web Interface
Linear Regression Model

Implementation

- Our project uses Google Cloud Platform to handle our data collection and analysis.
- Tools and Frameworks:
  - Python, Java, Hadoop/Spark, Flask, GUnicorn, Firebase (TBD), HTML/CSS/JavaScript
- The Python script runs on a flexible VM environment and the resulting data is then stored locally in the VM. This raw data will then be processed by the (non-persistent) Hadoop/Spark cluster and stored in another GCP container. The final data can then be stored in the VM or a GCP container.

Data Collection

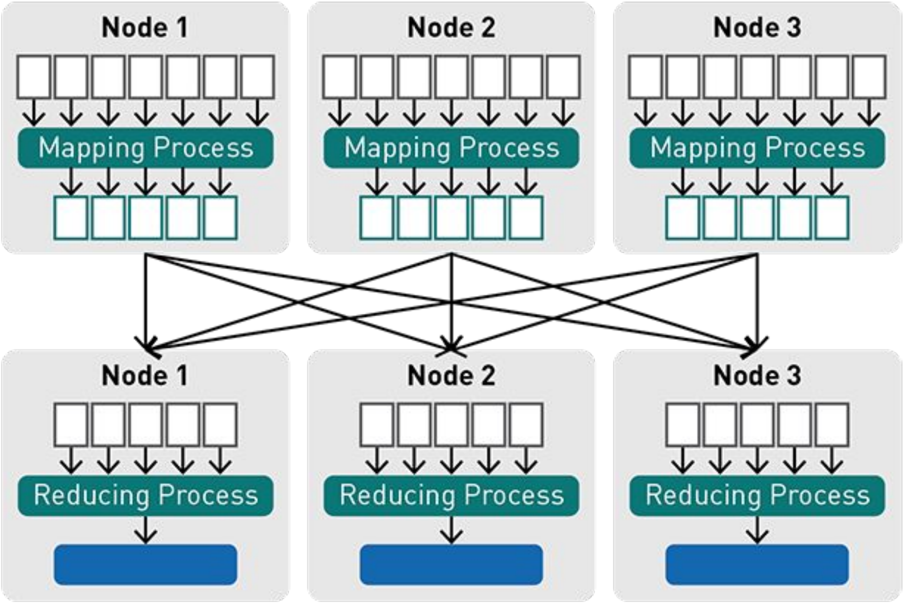
- Data collected from Twitter is a collection of responses to a given users tweet.
- Text responses allow us to gather trending words in responses to a recently posted tweet.
- Trending words can allow for a snapshot of discussion around a tweet.

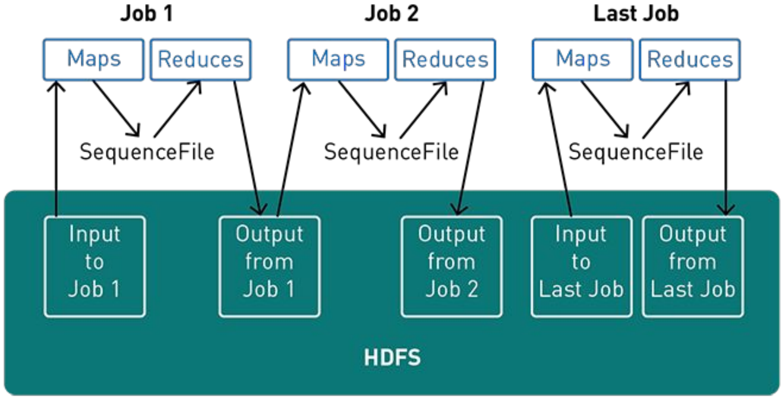
Hadoop and Spark

- Hadoop is useful for extremely large data sets or varied data sets (text files, images, etc.)
- Uses MapReduce and block processing
- Less costly due to the MapReduce model
- Does batch processing
- Highly secure

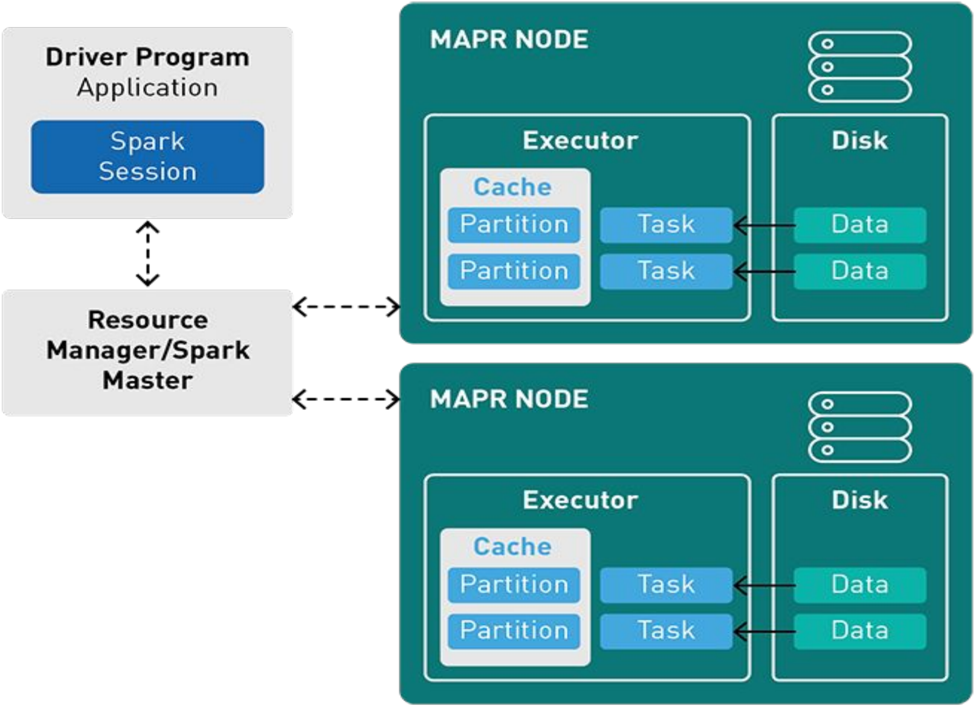
- Spark executes much faster by caching data in memory across multiple parallel operations
- Spark generally used for smaller/complex data sets
- Much more costly
- Does real time processing
- Less secure compared to Hadoop

How MapReduce works



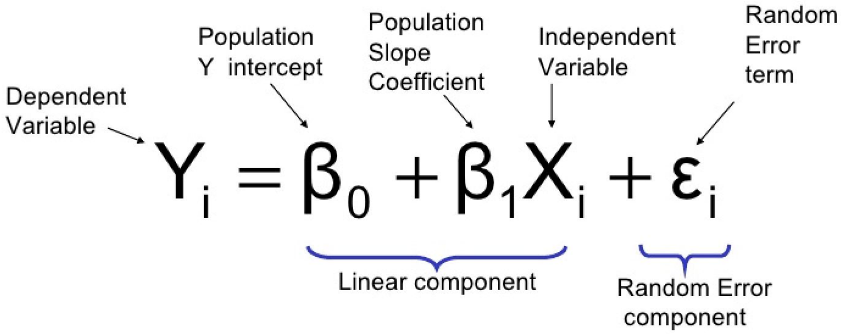


How a Spark application runs on a cluster



Linear Regression Model

- Attempt to find a relationship between independent and dependent variables
- Use R-Squared for analysis
- Able to find formula given the data



$$Y_i = \underbrace{\beta_0 + \beta_1 X_i}_{\text{Linear component}} + \underbrace{\varepsilon_i}_{\text{Random Error component}}$$

Dependent Variable

Population Y intercept

Population Slope Coefficient

Independent Variable

Random Error term

Demo

Web Interface

- The web interface is currently running a simple setup, using Flask as the web framework and Green Unicorn (Gunicorn) as the Web Server Gateway Interface (WSGI).
- This service is currently running in a flexible environment using Google Cloud’s App Engine. With the release of Google Cloud Run (similar to Amazon Lambda), this web service could run using their serverless platform, which would allow it to scale our stateless containers with demand. Since Cloud Run is serverless, it would abstract away all infrastructure management and all that would be needed is a static IP from Google to get everything up and running.

Challenges

- There are some issues that could be posed by the collection of Twitter data, such as:
  - User’s privacy in regards to deleted tweets
  - Keeping data secure in between processing (HDFS to Plaintext)
  - Ensuring only authorized users can access the data
- Some technical challenges faced by our team includes:
  - Sorting the contents of the data we acquire
  - Design and Research for the project, e.g. “What platform to use, which framework is best suited for our needs”.
  - Implementation of the Twitter script on our cloud servers and integrating the data with our Hadoop/Spark clusters for jobs.

Unfinished Tasks -> Finished Tasks

- The Python script can now run in the VM environment without issues, and that the data is being properly collected in a GCP bucket (compared to the VM's storage currently).
- Meaningful analysis of the collected data on the cloud-based cluster, rather than the test cases that were run locally
- An official web interface for the final data has been created

Sources used in Powerpoint

- Info about Carna botnet: http://census2012.sourceforge.net/paper.html
  - https://www.caida.org/research/security/carna/
- Google Cloud Run: https://cloud.google.com/run/docs/
- Hadoop/Spark: https://www.datamation.com/data-center/hadoop-vs.-spark-the-new-age-of-big-data.html
  - https://logz.io/blog/hadoop-vs-spark/
  - https://mapr.com/blog/spark-101-what-it-what-it-does-and-why-it-matters/
- Slide designs are from the CSUF EGCE 432 public templates on Google Slides