Steven Nodalo

6/28/2021

Open-Ended Capstone Step 4

Exploratory Data Analysis Write-up

This write-up contains insights from data explorations, as well as data storage plans for implementation.

# Archived Data

NEXRAD Archived dataset is organized in nested, zipped folders by location, date, and time. The data contained within the zipped folders is for visual analysis purposes only. No human-readable data is contained in the dataset.

NEXRAD data is stored under year > month > day > location. Under the location folder, there are several radar scan files contained for the specific day/location.

There are several tools available to access the NEXRAD data:

* nexradaws – A Python module designed to allow users to query and download NEXRAD radar files from AWS S3 storage buckets.
* Boto3 – A Python SDK that allows creation, configuration, and management of AWS services. This project can implement boto3 for AWS access.
* WeatherPipe – An analysis tool for NEXRAD dataset using MapReduce and EMR.
* TDS – Thredds Data Server contains NEXRAD data, as well as metadata for certain radar scans within the AWS S3 bucket. There are some drawbacks to implementing TDS (see below), which will be ultimate factors to whether TDS will be implemented within the project.
  + After implementing TDS within my Jupyter EDA notebook, TDS access is slow and may cause lag when extracting data.
  + Metadata provided by TDS is not provided for all radar scans.

## Thredds Data Server

Upon doing research and exploratory data analysis, I located Thredds data server, which provides metadata for AWS NEXRAD data. NEXRAD data contains only visual data of weather conditions. Thredds provides metadata that can be linked to NEXRAD data, which allows for a variety of different views for the data. The different views can be found in the [gallery here](https://unidata.github.io/python-training/gallery/gallery-home/).

* + siphon – A collection of Python utilities for downloading data from remove data services. Siphon is mainly used to access data within TDS.

# Live Data

NEXRAD Live dataset is a dataset that is continuously updated as it contains streaming data. After much exploration of the live dataset, I learned that this dataset is unnecessary for the project, as the live data is immediately transferred into the archived dataset. Additionally, due to the structure of the live dataset, it is difficult to implement into the project as it is not organized in a similar fashion to the archived data.

# Twitter Data

Twitter data will be used to enhance he NEXRAD dataset by linking the inclement weather with Tweets from affected individuals. This will allow scientists and analysts to observe what affected individuals were feeling, how the responded to the emergency, how difficult was the evacuation process, and what physical structures and/or evacuation procedures can be implemented in the future for high impacted areas.

There are several Python packages available for accessing Twitter data.

1. Tweepy – A Python library for accessing the official Twitter API.
   1. I was able to setup a Twitter developer account and obtain access to the official Twitter API.
   2. Because Tweepy utilizes the Twitter API, there are restrictions and limitations.
   3. Tweepy is easy to use and has thorough documentation
2. Twint – A scraping tool designed specifically for grabbing tweets.
   1. Because this is a web scraping tool that does not utilize the Twitter API, there are no limitations/restrictions.
   2. Twint offers very limited documentation.
3. search\_tweets\_api
   1. This is a wrapper for the official Twitter API, however it is outdated and offers minimal documentation compared to Tweepy.

Due to the extensive documentation offered by Tweepy, this project will be utilizing Tweepy within Python to access Twitter data.

1. Is the data homogenous in each column?

The NEXRAD data are homogenous in each column, as the data will serve as images and visuals. Data obtained from TDS is also homogenous, as it contains different views and angles of the same weather radar instance. The Twitter data is not homogenous in each column as some columns will have strings (such as the free\_text column depicting user tweets) while others will have integers (such as the date/time columns). Additionally, Twitter data may need to be cleaned due to unwanted punctuations and characters such as emoji’s and non-English characters.

1. How do you anticipate this data will be used by data analysts and scientists downstream?

This data can be used to analyze the severity of the inclement weather, as well as the impacts of lives of people who are affected by the inclement weather. Additionally, this data can serve in predicting which areas need updated evacuation guidelines by analyzing user Tweets of where affected individuals are evacuating to, the ease and ability to evacuate, and the number of affected individuals. With this information, scientists and analysts can suggest evacuation procedures for commonly impacted areas.

1. Does your answer to the last question give you an indication of how you can store the data for optimal querying speed and storage file compression?

The NEXRAD data is already compressed and organized by location and time, whereas the Twitter data will need to be extracted, cleaned, transformed, compressed, then linked to the NEXRAD data. Optimal querying can occur by storing Twitter data with relevant NEXRAD data. This can be done by separating Twitter data by time, location, and possible keywords that are relevant to the inclement weather. After Twitter data is separated, it can be stored within the compressed files along with the NEXRAD data.

Specifically, NEXRAD data is stored in year > month > day > location format. This project will obtain all Twitter data pertaining to the specific location/day event, compress all relevant Twitter data and store it within the same NEXRAD folder.

1. What cleaning steps do you need to perform to make your dataset ready for consumption?

The Twitter data will need to be cleaned by separating all data from the Tweet accordingly. Twitter post time and date will need to be uniform with the NEXRAD data’s time and date structure. Full text from the Tweets will need to be cleaned by removing any unnecessary characters, removing any non-English characters and words, and ensuring that all strings within the full text are separated into a list object for optimal querying. To properly store the strings into lists, removal of punctuations and white spaces will be necessary. Another possible form of data that may need to be stored are Twitter likes that are accrued per tweet.

1. What wrangling steps do you need to perform to enrich your dataset with additional information?

To enhance my NEXRAD dataset, I will need to properly structure and clean the Twitter data according to the structure of the NEXRAD data. Pipelines can be built which extract Twitter data based on time/date and possible location. Additionally, keywords can be used to detect and link Twitter data to NEXRAD data. Once Twitter data is extracted, it will need to be compressed, and placed into the compressed NEXRAD file which contains images of the weather event.

NEXRAD data is separated by station\_id, which is not very useful. Each radar station within the United States has a station ID. Since there is a limited number of stations, I can manually lookup the geographic coordinates of each station and save it into the metadata. This will allow me to link Twitter data via geocode with a 50 mile radius of each station.