Analysis of UFO Sightings Data and Unintended Consequences

Soravis Taekasem
taekasem@usc.edu

Surasit Prakunhungsut
prakunhu@usc.edu

Theerapat Chawannakul
tchawann@usc.edu

Yifan Wu
ywu352@usc.edu

Abstract

In this research, we explore several aspect of UFO sightings data using various tools and application including data extraction tool like Apache Tika and utilize similarity metrics by Tika-Similarity also apply some machine learning technique using Numpy, SciPy, Scikit-learn and Pandas library. Additionally, visualizing the data using framework like D3.js and Mathplotlib. Moreover, incorporate with some publicly available datasets to infer some potential insights.

1 Introduction

First and foremost we quickly dismiss the idea that these UFO sightings data were recorded from an observation of an alien life but more likely to be a misapprehend of an exotic aircraft, drone or rocket. To figure out how the UFO sightings are influenced by other factors, we aggregate it with datasets of U.S. major airports and military bases to analyse and see if these sightings are in a close proximity or further away from any particular location.

Furthermore, we incorporate the weather data and evaluate the weather condition of a sighting location at a certain date and time. Specifically, the visibility of the sky on that particular day. Finally, we use U.S. population demographics to investigate the relation between the population in each area to the sightings.

2 Datasets Observation

In this assignment we examine different kind of Multipurpose Internet Mail Extensions (MIME) type discuss in class. Thus, we mainly 3 top level MIME types including text, application and multipart.

2.1 UFO Sightings Dataset

UFO sighting dataset from infochimps.org [1] is a tab separated values (TSV) format (MIME type: text/tab-separated-values) with 6 fields including sighted_at, reported_at, location, shape, duration, description.

2.2 U.S. Airport Dataset

U.S. airport dataset from AcckiyGerman [2] is a comma-separated values (CSV) format (MIME type: text/comma-separated-values) with 13 fields including ident, type, name, coordinates, elevation_ft, continent, iso_country, iso_region, municipality, gps_code, iata_code, local_code.

2.3 U.S. Military Bases Dataset

U.S. military bases dataset from U.S. Department of Transportation [3] is a multipart content type format (MIME type: multipart/mixed) consist of 3 main part:

- Keyhole markup language (KML)
- Comma-separated values (CSV)
- Shapefile
  - Shape format (SHP)
  - Shape index format (SHX)
  - Attribute format (DBF)
  - Projection format (PRJ)
  - Code page (CPG)

2.4 Weather Dataset

Weather dataset from National Centers for Environmental Information [4] is a JavaScript object notation (JSON) format (MIME type: application/json) with 5 fields including station_identifier, date, observation_type, observation_value, observation_time.

2.5 National Population Dataset

National population dataset from United States Census Bureau [5] is in a tab separated values (TSV) format (MIME type: text/tab-separated-values) with 8 fields including year, fips_state, fips_county, age, race, sex, ethnic, population.

3 Implementation

In this report we present our approach of analyzing UFO sightings data by combining with four other publicly available datasets. In figure 1 shows the overall

process of our implementation. Broadly, our operation works in three stages, namely data processing, data analysis and data visualization.

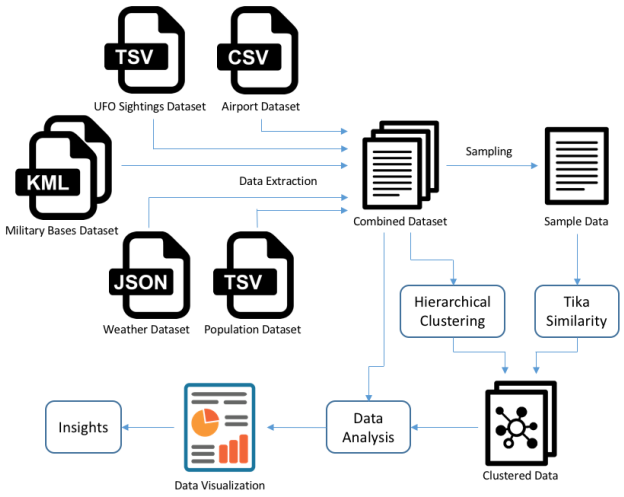


Figure 1: Process flow diagram

Initially, in data processing stage we start with extract the information from each dataset and aggregate them together to form the combined dataset. Additionally we sampling a set of data and use it as an input for similarity test.

Secondly, during data analysis stage we evaluate the combined dataset with three method which are query processing, clustering and also perform similarity test analysis using Tika-Similarity. Specifically, compare their similarity metrics such as edit distance, cosine similarity and jaccard similarity.

We then use several visualization tools and technique to present the clustered data in a meaningful way. For example, we use D3.js with scikit-learn library to visualize cluster dendrogram and D3.js with GeoJSON to create choropleth map.

4 Data Processing

Our data came from several sources, to analyze these data it need to be clean and organize in a certain manner. The objective of this stage is to prepare the proper input for data analysis stage.

4.1 Data Extraction

The UFO dataset contains many interesting features that provide information about UFO sightings. In order to generate an insightful final dataset for analyzing, we joined this dataset with other datasets that we observed. Once we got the final and combined dataset, we can then extracted only necessary features and do the analyzing.

4.2 Data Joining

From UFO dataset, two features that play an important role are "sighted_at" and "location" features. However, the location feature is hard to interpret as it is provided in text format. We used GeoPy python library to query for latitude and longitude by passing "location" feature as a parameter. Then, we use the acquired coordinates and sighted_at to join with these datasets in the following orders:

- U.S. airport dataset (combine_airport.py)
  We used latitude and longitude features to join with the airport dataset. For each sighting, we searched for the airport which is closest to that sighting. This is done by calculate the distance between every airport and the sighting using Great Circle algorithm. Then, we extracted "airport_name, airport_type" features including the calculated distance of airport to a UFO sighting.

- U.S. military base datasets (combine_military.py)
  The joining process with military base dataset is similar to U.S. airport dataset, we extracted "military_component, military_site, military_area" to final dataset. Also, we compared the coordinates in military dataset with UFO dataset and created a new feature "dist_to_military_base" to final dataset, which means the distance from a UFO sighting to nearest military base in miles.

- Weather datasets (combine_weather.py)
  The weather dataset contains features dayOfYear, weather (including rainy, snowy, foggy, windy, dusty), latitude, longitude. We extracted "weather" to final dataset. This dataset is joined with the UFO dataset by picking the nearest coordinates to the coordinates with UFO sightings, and picking the same day with UFO sightings "sighted_at" field at the same time.

- Population datasets (combine_population.py)
  The population dataset contains features "year, countyID, state, countyName, population, latitude, longitude". We extracted "countyName, state, countyID, population" as value of key "county" and added it to final dataset.

4.3 Data Sampling

For the initial input dataset, UFO dataset of 61,393 records. After we joined this dataset with other datasets within the range of 10 years(1990-1999), only 7,730 records remained. We used these 7,730 records for clustering. Further, we tried to run similarity test between each pair of files, unfortunately, Tika Similarity is not powerful enough yet to handle that huge amount of

files. To overcome this limitation, we randomly picked 20 records per year. As a result, we got 200 files for performing similarity test.

5 Data Analysis

After we combined all datasets together and extracted only the features that we wanted. Then we can do the following:

5.1 Query Processing

To explore for more insightful information of UFO dataset, we created scripts to count the number of UFO sighting based on each interesting extracted features. These features are what our team think are relevant to occurrence of UFO sighting.

- Distance to Airport (calculate_distance.py)
- Distance to Military Base (calculate_distance.py)
- Population of sighting county (calculate_population.py)
- Weather type (dusty, rainy, foggy, snowy, windy) (calculate_weather.py)
- FIPS code of sighting county (calculate_fips_code.py)

5.2 Similarity Test

After we combined all datasets together and extracted only the features that we wanted. We then go through the similarity test phrase in order to find similarities between each UFO sighting. We run three main similarity test algorithms,

- Jaccard Distance (value-similarity.py)
- Edit Distance (edit-value-similarity.py)
- Cosine Similarity (consine_similarity.py)

All given files for similarity test above using Tika to parse "metadata" field out from each file. This process consumes a lot of time and is not needed in our lab. We enhanced the python files to extract our features out of each file and use them as a feature set instead of the original metadata feature set. Thereafter, Jaccard Distance creates the "value-similarity-score.txt" file and both Edit Distance and Cosine similarity generate CSV files. These following files are then used later in the clustering phrase.

5.3 Clustering

With data from the combine feature dataset (7.7K rows) and data from each similarity test, we are ready for the clustering phrase. This phrase allows us to extract and reveal information hidden inside of our dataset.

We used K-Means Clustering to cluster the combine feature dataset with the K value of 5. For the similarity tests, we used two main algorithms for similarity clustering.

- Cluster Viz
  - cluster-scores.py for Jaccard Distance
  - edit-cosine-cluster.py for Edit Distance and Cosine Similarity

- Circle-packing Viz
  - circle-packing.py for Jaccard Distance
  - edit-cosine-circle-packing.py for Edit Distance and Cosine Similarity

Above python files are the part of Tika-similarity and are used to cluster the similarity result files. We enhanced both Cluster Viz and Circle-packing Viz python files to use our features instead of the default metadata feature set.

As the result of running the clustering files, K-means and Cluster Viz created the "clusters.json" file and Circle-packing Viz generated the "circle.json" file. These JSON files contain information of the file clustering that can be used to visualize the clusters.

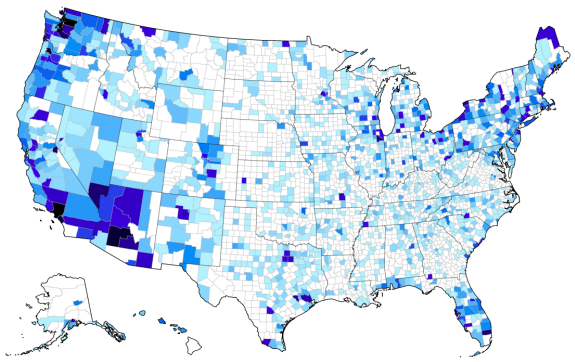


Figure 2: UFO sighting density

6 Data Visualization

Our data came from several sources, to analyze these data it need to be clean and organize in a certain manner. The objective of this stage is to prepare the proper input for data analysis stage. Some of the visualization result are shown in figure 2, 3 and 4.

For the Cluster Viz, we ran cluster-d3.html file that utilize the "clusters.json" from the clustering phrase and draw the clustering dendrogram using D3 javascript library. We can run dynamic-cluster.html file to generate an interactive dendrogram clustering.

For the Circle-Packing Viz, we ran the circlepacking.html file that utilize the "circle.json" and generate the clustering dendogram and dynamic-circlepacking.html to construct the interactive dendrogram clustering.

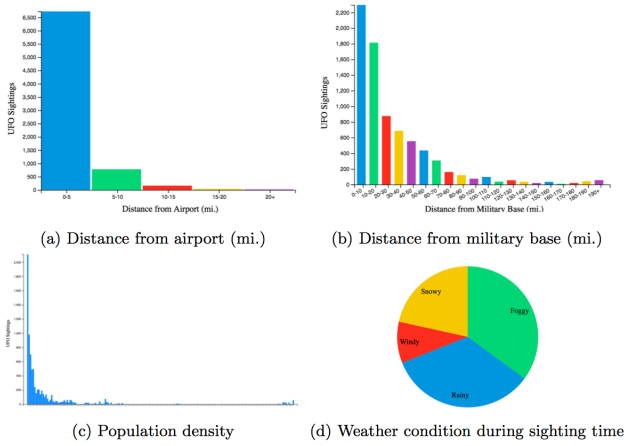
7 Results

7.1 Similarity Metrics

Jaccard distance/similarity output file revealed a very interesting result where all files in our lab received the same similarity score. We believe that this happens because our files contains the same features. Unlike the real-world web page files that have different features in each file, our file contains exactly equal amount of features.

Edit distance/similarity result is quite hard to analyze without any clustering. The similarity score are mostly scattered so we cluster them into four groups with 25% interval each. The result clusters show in figure 4(b) and figure 4(c), two big clusters where the first interval is 25%-50% and second interval is 50%-75%. When we looked closely to these two clusters, we found out that most data are packing between 40%-50%. So we conclude that in this lab, Edit distance is not a great indicator to follow.

Cosine distance/similarity result showed a very high similarity of all file where min-similarity score is 82% and most files has 99% similarity score. This makes sense because we add features to support the reason of seeing UFOs. For example, the distance to the closest airport and weather conditions like raining and foggy.



(a) Distance from airport (mi.)

(b) Distance from military base (mi.)

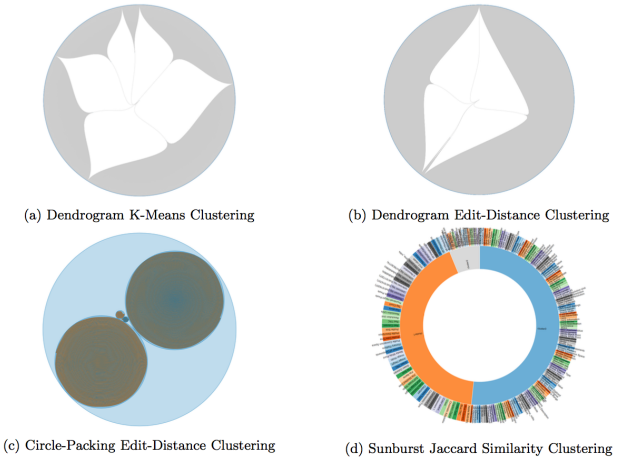
(c) Population density

(d) Weather condition during sighting time

Figure 3: Visualization

7.2 Insights

By querying and clustering our dataset, we obtained many interesting results as follows, UFO sighting happens more than 95% of a time when the sighting area is within 25 miles radius from the airport. And when we visualize UFO sighting density to airport location, they produce an almost identical chart (figure 2). This is the most significant observation in our dataset. Likewise, military base datasets also show the same trend more than 70% of observation fall within 40 miles radius.



(a) Dendrogram K-Means Clustering

(b) Dendrogram Edit-Distance Clustering

(c) Circle-Packing Edit-Distance Clustering

(d) Sunburst Jaccard Similarity Clustering

Figure 4: Clustering

8 Conclusion

From the datasets that we gathered, we were able to extract correlated information about UFO sightings by using multiple datasets and join them together. Moreover, by visualizing the combined datasets and running similarity tests, we come up with many insights about the factors that may related to occurrence of UFO sighting. Ultimately, we explore several visualization diagram such as sunburst in figure 4d, see more charts in a project directory.

References

[1] Infochimps.org, UFO Sightings Datasets
`http://www.infochimps.com/`

[2] AckiyGerman, Airport Datasets
`https://github.com/datasets/airport-codes`

[3] U.S. Department of Transportation, Military Bases Datasets,
`https://osav-usdot.opendata.arcgis.com/`

[4] National Centers for Environmental Information, Weather Datasets,
`https://www.ncdc.noaa.gov/`

[5] United States Census Bureau, National Population Datasets,
`https://www.census.gov/programs-surveys/popest/data/data-sets.html`