Homework: Large Scale Data Extraction and Analysis for Bigfoot
Due: Friday, April 5, 2024 12pm PT

1. Overview

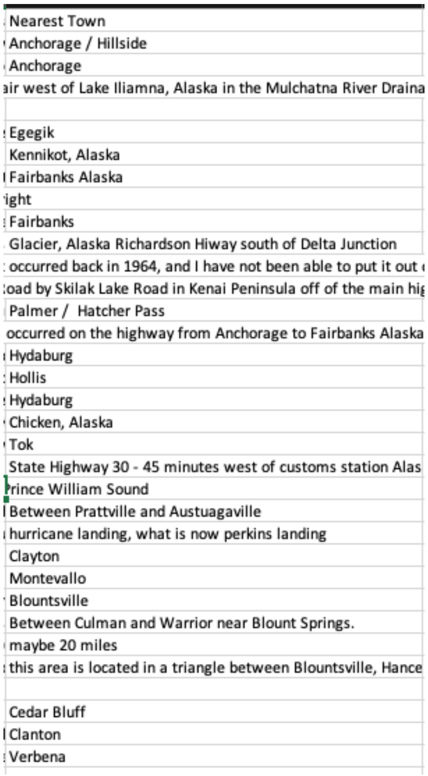
the man approached his home walking on all fours in the manner of an ape. He was about five feet tall , and had hair all over his body, He was unclothed .

Figure 1. Images generated using ImageFX from Google based on descriptions from one of the BFRO sightings.

In the second assignment, you will build upon the work that you did in assignment 1 in identifying additional explanations and features that helped you understand the BFRO data, and more information about Bigfoot sightings that could help you validate and verify them and their associated Class as identified by BFRO. You added features that correlated visitation in national parks, you added features that joined Weather datasets so you could explore the phenomena and associated physical properties of the sighting. You cleaned and identified the appropriate count of witnesses. Further, beyond these new columns, you were asked to select 3 datasets of 3 different top level MIME types and for each dataset, add 3 new features (total 9 new features) to your data. At the end of the new you added 14 new features to the data exploring the unintended consequences of Big Data, and the five 5 V's.

In this assignment you will focus on large scale content extraction and data science by exploring parts of the BFRO dataset left unexplored in assignment 1. You will explore the unique properties of locations in the dataset by using a GeoParser tool developed in the USC Information Retrieval and Data Science (IRDS) group.

There are plenty of **locations** mentioned in sighting observation description and associated text. As we have discussed in class during the advanced extraction lectures, as well as the metadata lectures, and clustering lectures, and as we will talk about during the Named Entity Recognition (NER) lecture, it is possible to use machine learning and natural language processing (NLP) to scan text and extract locations. For example, as you can see from the Nearest Town field in the data, you could input the text based information from nearest town and receive a coded Lat/Lng geolocation back.



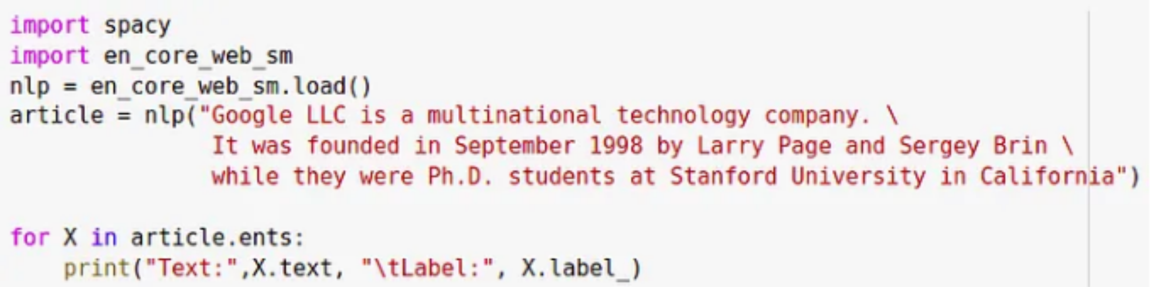
Nearest Town
Anchorage / Hillside
Anchorage
air west of Lake Iliamna, Alaska in the Mulchatna River Draina
Egegik
Kennikot, Alaska
Fairbanks Alaska
ight
Fairbanks
Glacier, Alaska Richardson Hiway south of Delta Junction
occurred back in 1964, and I have not been able to put it out
oad by Skilak Lake Road in Kenai Peninsula off of the main hig
Palmer / Hatcher Pass
occurred on the highway from Anchorage to Fairbanks Alaska
Hydaburg
Hollis
Hydaburg
Chicken, Alaska
Tok
State Highway 30 - 45 minutes west of customs station Alas
rince William Sound
Between Prattville and Austuagaville
hurricane landing, what is now perkins landing
Clayton
Montevallo
Blountsville
Between Culman and Warrior near Blount Springs.
maybe 20 miles
this area is located in a triangle between Blountsville, Hance
Cedar Bluff
Clanton
Verbena

Figure 2. Nearest town field from the BFRO sightings that you could input into the GeoParser tool.

Location names like "Prince William Sound", "chicken, Alaska", and so on, can be geocoded, and their corresponding latitude and longitude can be extracted using a tool developed by the USC Data Science Group called GeoTopicParser. The tool can take some text, and then perform an analysis using the Geonames.org database, and custom NLP and NER parsing code, using Tika, generating output that looks like the following if the text contained in the post mentioned, e.g., "China":

```
[
   {
      "Content-Type":"application/geotopic",
      "Geographic_LATITUDE":"39.76",
      "Geographic_LONGITUDE":"-98.5",
      "Geographic_NAME":"United States",
      "Optional_LATITUDE1":"27.33931",
      "Optional_LONGITUDE1":"-108.60288",
      "Optional_NAME1":"China",
      "X-Parsed-By":[
         "org.apache.tika.parser.DefaultParser",
         "org.apache.tika.parser.geo.topic.GeoParser"
      ],
      "X-TIKA:parse_time_millis":"1634",
      "resourceName":"polar.geot"
   }
]
```

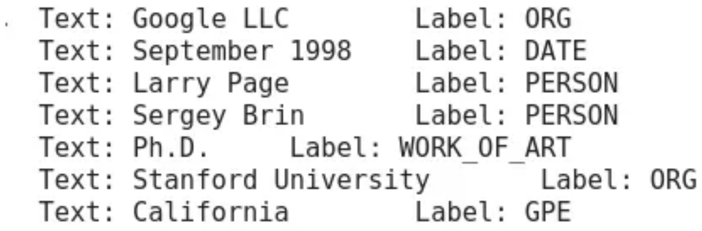
One last thing that we can do with the text, illustrating large scale content analysis is to run machine learning approaches on the BFRO sightings to determine other named entities in the sightings text. We can do this in assignment 2 by leveraging machine learning, and the SpaCY tool is a named entity recognition library that will examine text and identify all of the associated entities present in content in the text.



```
import spacy
import en_core_web_sm
nlp = en_core_web_sm.load()
article = nlp("Google LLC is a multinational technology company. \
               It was founded in September 1998 by Larry Page and Sergey Brin \
               while they were Ph.D. students at Stanford University in California")

for X in article.ents:
    print("Text:",X.text, "\tLabel:", X.label_)
```

Output



```
Text: Google LLC        Label: ORG
Text: September 1998    Label: DATE
Text: Larry Page        Label: PERSON
Text: Sergey Brin       Label: PERSON
Text: Ph.D.     Label: WORK_OF_ART
Text: Stanford University       Label: ORG
Text: California        Label: GPE
```

Figure 3. The SPaCY tool and named entity recognition. Notice in the top portion an article is input into the tool in Python and the bottom portion is the output from SpaCY.

However, we're not stopping at the text, and in this assignment, you will generate an image for each associated BFRO sighting. You will accomplish this by leveraging a Generative AI tool such as ImageFX from Google and providing it with the Observation text and experimenting to generate an image for each BFRO sighting. Once you have the image associated with the sighting, you can then use a machine learning based Image Captioning algorithm called Show & Tell originating from Google to automatically caption and generate text features about the Pixstory posts and use these captions as additional features to describe the post. We will leverage two easy to use Tika Docker files to identify objects present in an image and to generate a textual (human readable) caption for the image. Both of these Docker Files are available in Apache Tika and they leverage Machine Learning and Deep Learning extraction techniques in particular Google's Tensorflow technology and custom Deep Learning models built in the USC IRDS group. You can see some examples of the Image Captioning and Image Object identification in action below in Figure 4a-c showing 3 automatically generated labels (with only generic training). We will integrate this Tika capability and generate labels and text captions for your BFRO sightings images.

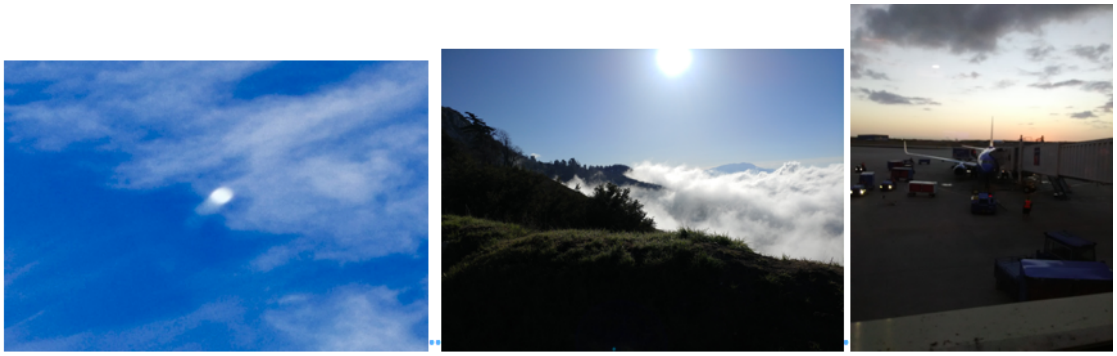


Figure 4: a) a light/orb shown in the daylight; b) an orb present against a mountain background; and c) an orb in a cloudy sky.

a plane flying in the sky over a field	*a view of a mountain range with a mountain in the background*	*an airplane is parked on the tarmac at an airport*

Machine Generated Labels for a). b). c)

The combination of these techniques will allow you to apply knowledge gained from the Parsing/Extraction Lecture, the lectures on advanced extraction (including Deep Learning and Metadata), and also topics discussed including Large Scale Content Extraction. In particular, please consider techniques discussed in class to embark on this assignment.

2. Objective

The objective of this assignment is two-fold. First, you will expose the richness of the sightings, and generate accurate, geolocated Lat/Lng that can be used to map and understand your sightings data. You can identify location names using the GeoTopicParser and geocode the associated locations in the sightings and then finally you can run SpaCY to automatically classify the toxicity of the posts. The location names will also aid you in comparing e.g., sightings, associated features, and information together from your prior assignment 1 work. In addition, using SpaCY will provide you with entities that you can use to further classify and study your BFRO data.

In addition, the other objective of this portion of the assignment is to leverage the richness in the underlying images you generate for each BFRO sighting, along with large scale machine learning and data science, to generate text from the images (a caption), along with the list of objects present in the imagery. Both will provide additional features with which you can examine the sightings, and compare: are these images representative of the sighting and something that provides a visual aid to better understand the report? You will explore these questions in assignment 2.

The assignment specific tasks will be specified in the following section.

3. Tasks

1. Generate a copy of your TSV v1 dataset. Call it "v2" or something similar. You will add your new columns for
    a. GeoTopic name, along with associated lat/lng
    b. Columns for all generated SPaCY named entities
    c. A column pointing on disk to your generated AI image for the report.
    d. Image Caption generated by Tika's Show & Tell caption generator
    e. Detected objects in the image using Tika's Inception Rest service
2. Download and install Tika Python using PIP and the instructions at http://github.com/chrismattmann/tika-python
3. Install GeoTopicParser using the instructions here https://cwiki.apache.org/confluence/display/tika/GeoTopicParser
    a. The result of this should be the Lucene GeoGazetteer REST server running as specified here: https://github.com/chrismattmann/lucene-geo-gazetteer
    b. You can connect the GeoGazetteer to Tika-Python using the instructions here: https://github.com/chrismattmann/tika-python#changing-the-tika-classpath
4. Install SpaCY using PIP and the instructions here: https://spacy.io/usage
5. Use a Generative AI image generator service such as any of the following: ImageFX from Google, Imagine from Meta, Midjourney, DALL-E 3 from OpenAI
    a. Leverage the text from your sighting columns, such as Observed, or Witness count, etc.
    b. Combine that text together into a caption
    c. Input the caption into the Image Generation service to receive the generated image
    d. Note that you can use the API calls to these services to script/automate your process
6. Install Tika Image Dockers and generate captions for your BFRO generated images from step 5
    a. To access the images, use a local file URL from your generated BFRO sightings
    b. Install Tika Dockers package for Image Captioning and Object Recognition
        i. git clone https://github.com/USCDataScience/tika-dockers.git and https://hub.docker.com/r/uscdatascience/im2txt-rest-tika
        ii. Read and test out: https://cwiki.apache.org/confluence/display/TIKA/TikaAndVisionDL4J
        iii. Read and test out: https://github.com/apache/tika/pull/189
    c. Iterate through all the BFRO sightings images and add the generated image caption and the detect object(s) column to your dataset
7. Iterate through all the BFRO sightings and then run Tika GeoTopicParser and extract out Location name, including Lat/Lng based on your text fields for the sighting
    a. Write a Python program to do this

    b. Add the new column(s) to your dataset

8. Run SpaCY on all the BFRO sightings and extract the associated entities present in the sightings text
    a. Add these columns and scores to each of your sightings in your new dataset

4. Assignment Setup

4.1 Group Formation

You can work on this assignment in groups sized at **minimum 2, and maximum 6**. You may reuse your existing groups from discussion in class. If you have any questions, contact the TA via his email address with the subject:
DSCI 550: Team Details.

5. Report

Write a short 4-page report describing your observations, i.e. what you noticed about the dataset as you completed the tasks. For example, the following questions are of interest.

1. Are there any sightings correlations by location in the posts?
2. Are there correlations between the national parks, and/or entities identified with locations?
3. Do the Entities provide any further context about the sighting? What about witness count? Does it allow you to further validate the witness count?
4. Do the image captions accurately represent the image?
5. Are the identified objects present in the image described in the original sighting and/or the generated caption?
6. Are there any specific trends you see in the text captions or identified objects in the image media?

Also include your thoughts about the ML and Deep Learning software like GeoTopicParser, SpaCY, Tika Image Captioning, etc. – what was easy about using it? What wasn't?

6. Submission Guidelines

This assignment is to be submitted ***electronically, by 12pm PT*** on the specified due date, via Gmail dsci550sp24@gmail.com. Use the subject line: DSCI 550: Mattmann: Spring 2024: EXTRACT Homework: Team XX. So if your team was team 15, you would submit an email to dsci550sp24@gmail.com with the subject "DSCI 550: Mattmann: Spring 2024: EXTRACT Homework: Team 15" (no quotes). **Please note only one submission per team**.

- All source code is expected to be commented, to compile, and to run. You should have at least a few Python scripts that you used to join three other datasets, and what you used to extract additional features.

- Use relative paths {not absolute paths} when loading your data files so that we can execute your script/notebook files without changing everything.

- If using a notebook environment, use markdown cells to indicate which tasks/questions you are solving.
- Include your updated dataset TSV. We will provide a Dropbox or Google Drive location for you to upload to {you don't need to attach it inside the zip file}.
- Also prepare a readme.txt containing any notes you'd like to submit.
- If you used external libraries other than Tika Python, you should include those jar files in your submission, and include in your readme.txt a detailed explanation of how to use these libraries when compiling and executing your program.
- Save your report as a PDF file (TEAM_XX_EXTRACT.pdf) and include it in your submission.
- Compress all of the above into a single zip archive and name it according to the following filename convention:

  TEAM_XX_DSCI550_HW_EXTRACT.zip

  Use only standard zip format. Do **not** use other formats such as zipx, rar, ace, etc.
- If your homework submission exceeds the Gmail's 25MB limit, upload the zip file to Google drive and share it with dsci550sp24@gmail.com.

When submitting, please organize your code and data file as the directory structure shown:

```
Data
dataset1 {leave it empty for now}
Source Code
     script/notebooks
Readme.txt
Requirements.txt
```

Important Note:

- Make sure that you have attached the file the when submitting. Failure to do so will be treated as non-submission.
- Successful submission will be indicated in the assignment's submission history. We advise that you check to verify the timestamp, download and double check your zip file for good measure.
- Again, please note, only **one submission per team**. Designate someone to submit.

6.1 Late Assignment Policy

- -10% if submitted within the first 24 hours
- -15% for each additional 24 hours or part thereof