TEAM_02_CSCI599_HW_BIGDATA_Report

Input File: ufo_awesome_FINAL_OUTPUT_v2.tsv (renamed the version 1 file)

Output file: ufo_awesome_FINAL_OUTPUT_v2.tsv (appended to the above input file)

Data Preparation and Cleaning

We started with running the provided ocr-pipeline.sh bash script (https://gist.github.com/chrismattmann/a5031c317bad35ca30cec7b9decd51a5) on the British UFO pdf files. We had to install ghostwriter software along with the mentioned software. We modified the script and explored some options of imagemagick to fix errors like 'spp not in set {1,3,4}' using '-background white -alpha Off' option and improve the OCR quality for files which were not giving any text output. We did border removal by using -border, -fuzz, -trim options.

The most important pages were "Reports for Unidentified Flying Objects", from where we extracted the information about UFO sightings, like Date, Time and Duration of Sightings, Description of object, where the observer was, how was it observed, what were the nearby objects or the weather conditions when the observation was made and were there any other witnesses to the particular UFO sightings.

Observations made about the extracted dataset:

1. While implementing OCR on the pdf files, most of the data we initially received was improper and most of it was just garbled text.
2. Since the original documents are old and little bit tattered, some the scanned pages have low visibility due to faded text and ink blots. So, no useful data is being extracted from them.
3. Some of the records are handwritten in cursive, which Tesseract is unable to read.
4. We considered only those pages in the PDF which contains the keywords "REPORT OF UNIDENTIFIED FLYING OBJECT".

Parsing OCR extracted data into TSV:

We used the following British UFO files dataset: https://www.dropbox.com/sh/bwzhuhigz222rwr/AADNTCqrTdtD78sXdWrEHUxsa?dl=0

There are 8 British UFO pdf files. The pages of interest in the pdf files contained the keywords "FLYING OBJECT". We ignored the pages which did not contain the mentioned keywords. Since the extracted text from OCR scanning contained few garbled characters, we decided to apply regex to look for the following keywords:

1. Flying or aerial (if page is of interest)
2. Date & Sighting (for Date of Sighting)
3. Mins, Secs, Hours, Still there (for Duration)
4. Description (for Shape)
5. Position (for Location)
6. How Observed + Direction (for Description)
7. Receipt (for Date of Report)

We followed the following steps to extract data from scanned PDF files and parse them into TSV:

1. We split each PDF file into individual pdf files with one page per file.
2. We converted the individual PDF files into their respective tiff images using imagemagick library. Since the scanned PDFs have low visibility and have a lot of noise, we applied few image

enhancement features of imagemagick like noise reduction, darkening the text and redefining the borders.

3. We then used Tesseract to extract the data from TIFF images into corresponding text files. Since the output text had different kind of spelling errors, garbled text –we tried to use autocorrect lib from python for correcting spelling mistakes and enchant lib from python to identify if a word belongs to en-US dictionary. We could not fix them, as there were two or more words stitched together and we had no way to identify them as distinct words. Also, there were many non-ASCII characters in the extracted text. So, we had to manually edit some of the text files to correct the keywords we are looking for.

Image scraping from UFO Stalker using Selenium:

Scraping the images from the ufostalker website was a bit of a task because of its not so scraping-friendly website design. Since the pages were not hyperlinks but angular-induced function calls, going to a page towards the end involved traversing all the intermediate pages in between which ate up into too many calls being made. Another hiccup while scraping was the issue of IP blacklisting for which we made use of a VPN and a good time lag of about 3 seconds in between calls.

Upon analysis, we discovered that the event ids ran from 1 to 91148 and we could instead parse them based on these event ids. A simple regex based script was written which then scraped all image urls from the ufostalker website.

Object Recognition and Image Captioning using Docker:

We worked on about 5400 images which corresponded to the 2800 unique sightings. For this assignment, we are considering each image as a unique sighting. The sightings can be aggregated based on the unique identifier present in the beginning of filename

Features extracted from scraped Images.

a) Object recognition using Inceptionv4 docker: - The Objects discovered by the Inception v4 model mostly represent the surroundings rather the object of interest in the image. Some recurring objects that were getting recognized are barn, lakeside, parachute, aero plane, balloon, church. Mostly these corresponded to the pictures clicked in an open setting such as an open field or blue sky. A lot of pictures which had a part of a big building were getting classified as a church. Because mostly the “sighted UFO” was small or just a flash of light, they were not getting recognized the Inceptionv4 Tensorflow model as it is trained on rather clear pictures which have distinct objects to be recognized.

b) Image Captioning using docker: - As Image Captioning also relied on the Inception model we could see the similar issues faced in object recognition. A lot of captions included the strings” with a sky background”, “in the middle of the forest”, “in a lush green field” etc. Image captioning mostly described the background view rather than the UFO object sighting.

c) Metadata Features: - On analyzing the UFO stalker images, we found that a lot of the image’s metadata contains features which would be useful to include and comply with the previous TSV v1 we populated in last assignment. Below is the detail of how those features were extracted. We used Tika’s AutoDetectParser to get the metadata for images. Initially we tried using URL stream directly to detect metadata, but it was neither able to capture the metadata nor able to detect the mime-type of images (All images were being classified as text/html). So, we decided to download all the images and then pass it to Tika’s Parser as a FileInputStream, which worked like a charm.

- **Date of Sighting**: - There were multiple date features present in the metadata such as "Date/Time Original", "Date/Time Stamp", "Date Created" etc. Thus, we analyzed the metadata of images to get hold of the most accurate timestamp. We came up with a priority order on the dates fields such as: - "Date/Time Original" > "GPS Date Stamp". Also, we ignored all the dates which had 00 or some default time set.

- **Geolocation**: - A lot of pictures clicked in last 7-8 years also had the GPS latitude longitude information, with majority of them were clicked from smartphones. The corresponding metadata features were of the form "GPS Latitude", “geo:lat”. We used geopy API to get the geographical location of sightings.

Insights from the joined dataset:

1. **What we noticed about the dataset?**
   The UFO Sightings dataset provided by British UFO files have more detailed description than the original dataset we received for Assignment 1. The details like what nearby objects could have been misunderstood as a UFO or what Meteorological conditions could have triggered the observation, can actually provide us with hints whether the observation was true or was just a deception.
   The image dataset: - The images from the UFO stalker were a lot of time blurry and had the problem of occlusion. In some cases, viewers had used image tools to circle or highlight the sighted UFO in image. In some cases, there were just aero planes. Some of the images were drawn by hand or were some other kind of representational image.

2. **What questions did the newly joined answer about UFO sightings previously unanswered?**
   Through OCR pipelining of British UFO files, we got to know about the precise location of the observer and what the observers’ activities were during the observation. Earlier UFO Sightings dataset just mentioned the city and state where the sighting occurred. Besides that, there were many new shapes of the UFOs described which gave more detailed information about the sightings as compared to the information given by UFO Sightings dataset in Assignment 1.
   For UFO stalker dataset, there was no straightforward question which we could think of that this dataset helped to answer.

3. **How well did the image captions accurately describe the UFO object types? What about the identified objects in the image?**
   Most the time image captions and object recognition were recognizing/describing the background of the picture rather than the object. In very few cases I could see where the plane/balloon/parachute/chute/volcano (for flash of light) words were used to describe the object detected in the sky.

4. **How well did the OCR work? What did we do to clean up the noise in the data?**
   Due to a lot of noise in the original British UFO files, some of the pages in PDFs produced garbled texts when OCR was operated on them to extract text. It was because most of the characters, even though were typewritten, had low visibility, blotted ink or had some dirt on the pages because the scanned files are very old. To clean up the noise, in order to extract better readable data, we played around with few features of magick (imagemagick). For example- we redefined the border color and border thickness, we defined the density of the generated image to maximum 300 dpi level, we introduced fuzz for color matching up-to 20% and trim to trim the borders of the identical color as the corners of an image.

5. **Of the incorporated British UFO Sightings, how many of them could also similarly be explained akin to the sightings from the first assignment?**
   The UFO Sightings dataset mostly contained UFO Sightings from United States, while British UFO Sightings, as the name suggests, contains UFO Sightings from area of United Kingdom. Though some of the data in UFO Sightings dataset from Assignment 1 contained some UFO Sightings from United Kingdom, we could not find much similarity between the sightings between the two

assignments. It is so because the dataset in first assignment was based on city and state location, while in 2nd assignment there is precise location of the observer (where he was when he sighted the UFO). Also, there are lot of inconsistencies in the format of date of sightings and report. The only similarity can be based on dates, if we parse them into the format the original UFO Sightings dataset had in Assignment 1.

6. **Were there any new object types introduced by British UFO Sightings?**
   We found some new and interesting object types or shapes like ‘helicopter disc shaped’, ‘milk bottle base’, ’wulcane’, ’A/C’, ’10 pence coin’ introduced by British UFO Sightings

7. **How well were the British UFO Sightings described? Was there a lot of missing data?**
   Some of the data from the British UFO Sightings dataset was well described, like what nearby objects could have been misunderstood as a UFO or what Meteorological conditions could have triggered the observation. But there was a lot of missing data like some date of sightings and report just specified the time of report, and in some places the format of date given was really hard to parse into a proper date format. There were many inconsistencies in how the report was filed. Also, some of the reports were handwritten and thus Tesseract OCR pipeline was unable to extract any readable data from those scanned pages.

8. **Of the UFO images, how many of the images actually generated image captions and/or objects that described the UFO and not just the background scenery?**
   Out of all the images I could see that about in 5% cases the object in the sky was detected to be a balloon/plane/chute/volcano.

9. **What was easy about OCR pipelining and Image Captioning and what was not?**
   The OCR Pipelining was very easy to use, as we just had to convert the PDF files to TIFF images (the highest resolution dpi images) and then extract the data using the tesseract command. But since our dataset was scanned images of very old documents, tesseract did not perform well on OCR extraction of data as the scanned images were noisy and had low visibility. This led us to play around with lot of image enhancement features of imagemagick, but for some images, it still was unable to extract data. Also, tesseract failed to extract data from the images in which the data was handwritten or poorly scanned. For Image captioning task, use of docker made it quite easy, but as discussed above, a lot of captions generated were not accurate as they were not able to detect/recognize the sighted UFO.

10. **Why we chose not to write a Tika Parser?**
    We studied about the Tika parser and found out how it works. Tika parser takes in a file and extracts the metadata and content from that file and then it returns a stream of data on which further parsing needs to be applied. Since we had to define the parsing function for TIKA in Java, we decided to write our own parser in Python because of the ease of use of the language. Also, since there were many inconsistencies in the extracted data (like spelling errors, non-ASCII characters), we decided to write our own parser in Python instead of Java so that we can make use of the extensive libraries of Python like autocorrect and regex.

NER Features added using Apache OpenNLP:

As part of the extra credit, we have added four extra NER features for first 61207 rows of our TSV dataset by running tikaNER tagger on the Description column. The corresponding four features are: -

1) NER_PERSON
2) NER_LOCATION
3) NER_ORGANIZATION
4) NER_DATE