**Lab Report**

Title: Comparing different Web API’s MNgeo, Google places and NDAWN

Notice: Dr. Bryan Runck

Author: Samikshya Subedi

Date: 10/9/2022

**Project Repository:** <if applicable weblink to public repository>

**Time Spent:** 40 hrs

**Abstract**

<Delete this text in light grey throughout>

250 words max. Clearly summarize the following major sections. Each gets one or two sentences.

**Problem Statement**

Compare and contrast the conceptual models for the Minnesota Geospatial Commons, Google Places and NDAWN API’s

Create Jupyter notebooks that can programmatically get data from each of these APIs.

Table 1. Data requirement for Conceptual models

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **#** | **Requirement** | **Defined As** | **(Spatial) Data** | **Attribute Data** | **Dataset** | **Preparation** |
| 1 | MNgeo dataset | Raw input dataset from Rice County | Geo coordinates | point | [Minnesota Geospatial Commons](https://gisdata.mn.gov/content/?q=help/api) | N/A |
| 2 | API | GPS Coordinates | Geocoordinates | point | [Google Places](https://developers.google.com/places/web-service/overview) | N/A |
| 3 | Weather data | Raw input dataset from NDAWN | Discontinuous Data | Temp, elevation, wind humidity | [NDAWN](https://ndawn.ndsu.nodak.edu//) | N/A |

**Input Data**

Table 2. Table showing purpose in analysis for different data sets.

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Title** | **Purpose in Analysis** | **Link to Source** |
| 1 | Land application site and public drainage systems | Raw data input for API Land application site and water public drainage system for spatially joining their co-ordinates | [Mn GeoSpatial Commons](https://gisdata.mn.gov/dataset/trans-roads-mndot-tis) |
| 2 | Nearby restaurant locations | Input latitude and longitude of nearby restaurants locations | [Google Places](https://developers.google.com/places/web-service/overview) |
| 3 | Monthly weather data | Input monthly weather data from NDAWN for comparing API to spatially join their coordinates | [NDAWN](https://ndawn.ndsu.nodak.edu//) |

**Methods**

**Calendar

Description automatically generated**

Figure 1: Dataflow diagram

1. Google places:

A personal API key, which can be easily obtained through Google's Maps Platform, is needed to programmatically access data from Google Places. A Jupyter notebook was made when the API key was obtained in order to collect information some restaurants close to Minneapolis. Longitude, latitude, and the search radius were all entered into a Python program along with the special API key. Name, Latitude, and Longitude are the three columns in the final output table of the pandas conversion of the output JSON file.

Graphical user interface, text, application, email, Teams

Description automatically generated

Figure 1: Python code from google maps platform nearby restaurant search

Graphical user interface, text, application, email

Description automatically generated

Figure 2: Converting JSON output to pandas for output table

1. NDAWN:

Use NDAWN website to collect the monthly weather data <https://ndawn.ndsu.nodak.edu/table.csv> and write a python code and insert gauge station, year, weather data type etc. After that feed CSV file as another input and get the final output in tabular form with information such as longitude, latitude, elevation, monthly normal air temperature, departure from monthly air temperature.

Graphical user interface, text, application, email

Description automatically generated

Figure 3: Figure showing python code and insert gauge station, year, weather data type

etc. After that feed CSV file as another input to get the final output in tabular form

1. Minnesota Geospatial Commons

Open url [https://resources.gisdata.mn.gov/pub/gdrs/data/](https://l.facebook.com/l.php?u=https%3A%2F%2Fresources.gisdata.mn.gov%2Fpub%2Fgdrs%2Fdata%2F%3Ffbclid%3DIwAR0SLuouMVv51j25bMwZd5nLBfOQu5VosBgJ4QNBZ9Cb02w0c5-8mF6ydFk&h=AT3kJhBjhVFBSCR30FzawtX8CBNvO5-9-hgwEpl9eh78CTaEAdYj-rixtE7-ipA6nU2Wy77Tqhp6RH2gWSDKxYEQas8feRCtQSVNVGnFl0oSYAKTNV2srs1BrHPnj5iJkqOp7XT8SZhnhRpF8peLOQ)" Download zip environment land application site, water public drainage system and shape file environment land application site. Import python modules and write required python codes as shown in the picture below and after that we assign the proj\_gdb and working folder from project folder respectively from C drive. Then arcpy env workspace is assigned. Export the feature using arcpy conversion and get the spatial references output. Finally input data frames and get the final output.

**Graphical user interface, text, application

Description automatically generated**

**Graphical user interface, text, application

Description automatically generated**

**Figure 4: Figure showing Python codes for Mn**

**Results**

This lab report contains Only the findings from informal conceptual models—Google Places, the North Dakota Agricultural Weather Network are included in this lab report (NDAWN).

**Google places:**

The list of restaurants within a 1500-meter radius of a chosen area is the outcome of using Google Places (44.97410038936722, -93.22920853198917)

Graphical user interface, text, application, email

Description automatically generated

Figure 4: Results showing output table with name latitude and longitude of nearby

restaurants.

**NDAWN:**

Graphical user interface, table

Description automatically generated

Figure 5: Figure showing NDAWN output table

**Results Verification**

Results is verified when screen the head of the table showing the merged attributes of Google Places, NDAWN and Minnesota Geospatial Commons

**Discussion and Conclusion**

I tried to learn Compare and contract different web API’s routines. It relates in Building an ETL pipeline with Open Source Tools in Esri’s Online and ArcPro Jupyter Notebook and integrate a two datasets via spatial join

**Self-score**

Fill out this rubric for yourself and include it in your lab report. The same rubric will be used to generate a grade in proportion to the points assigned in the syllabus to the assignment.

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Description** | **Points Possible** | **Score** |
| **Structural Elements** | All elements of a lab report are included **(2 points each)**:  Title, Notice: Dr. Bryan Runck, Author, Project Repository, Date, Abstract, Problem Statement, Input Data w/ tables, Methods w/ Data, Flow Diagrams, Results, Results Verification, Discussion and Conclusion, References in common format, Self-score | 28 | **28** |
| **Clarity of Content** | Each element above is executed at a professional level so that someone can understand the goal, data, methods, results, and their validity and implications in a 5-minute reading at a cursory-level, and in a 30 minute meeting at a deep level **(12 points)**. There is a clear connection from data to results to discussion and conclusion **(12 points)**. | 24 | **24** |
| **Reproducibility** | Results are completely reproducible by someone with basic GIS training. There is no ambiguity in data flow or rationale for data operations. Every step is documented and justified. | 28 | **23** |
| **Verification** | Results are correct in that they have been verified in comparison to some standard. The standard is clearly stated **(10 points)**, the method of comparison is clearly stated **(5 points)**, and the result of verification is clearly stated **(5 points)**. | 20 | **20** |
|  |  | 100 | **95** |