A close up of a map

Description automatically generated

Data Update and Manipulation on a Web Application for finding the shortest feasible route

for Alternative Fueling Vehicles

by

Jingchao Zhou

A Capstone Proposal Presented in Partial Fulfillment

of the Requirements for the Degree

MAS-GIS

ARIZONA STATE UNIVERSITY

April 2020

Table of Contents

Page

1. SCOPE OF WORK ……………………………………………………………………. 1

Background 1

Purpose 2

1. METHODOLOGY …………………………………………………………………….. 4

Data Collection and Analysis 3

Data Manipulation…………………………...………………………………...3

Optional Work…………………………………………………………………5

Challenges…………………………………………..…………………………6

1. TIMELINE……………………………………………………………………………...7
2. DELIVERABLES………………………………………………………………………8

5 REFERENCES ………………………………………………………………………….9

**Scope of Work**

**Background**

Fuel stations for alternative fueling vehicles are not as adequate as those for the regular vehicles, in terms of both the quantity and the support of online mapping tools that help the drivers to find feasible routes without running out of fuel. Finding available stations and planning an efficient route is crucial for the drivers who own the alternative fueling vehicles. To help with these issues addressed, Dr. Michael Kuby and his group created an online mapping tool for alternative fueling vehicle routing with range and refueling stations in 2014. This tool will help the drivers to find the shortest feasible path from the origin to the destination. The users can put in start and end location, driving range of the vehicle, initial fuel level, and travel type in the interface. Then the tool will generate the feasible shortest path with the stations for refueling. However, the data of the stations and pre-processed distance between all pairs of the station points have not been updated since 2014.

A close up of a map

Description automatically generated

Figure 1. An example of the shortest feasible path from Cleveland to Tempe generated by the mapping tool.

**Purpose**

The change of the alternative fueling station since 2014, including the hydrogen and GNC (compressed natural gas) station, may not be significant in terms of its quantity. However, the discrepancy of data will cause inaccuracy of the generated path and stations, then affect the user experience of the tool. For example, the user could be led to a nonexistent station because the station may not exist in 2020.

The project will include collecting data sets in 2020 for the AFV stations, using scripting to generate shortest distance pairs, and updating the user interface of the app. Due to the requirement of large computational power, the distance between all pairs of nodes is pre-generated by python programming and ESRI’s ArcGIS online geocoding service. I will need to regenerate the distance of the nodes because of the update of the station-location data will affect the result of the previously generated distance.

The capstone project will help to update the station data from 2014 to the current 2020 version. It will help me to apply the data management and programming knowledge I have learned in the MAS-GIS program, consolidate and review the skills of Python and SQL. I will also help to deepen my understanding of the relational database management system by using PostgreSQL and PostGIS extension that applies the SQL query language in a real-world project.

**Methodology**

**Data Collection and Analysis**

The station data of the hydrogen and CNG are from the website of Alternative Fuel Data Center: afdc. energy.gov. After downloading the 2020 data, I will compare the fields of the data and change of values of the relational database, so I can decide what I can change for the previous scripts. The current tool allows the user to choose the origin and destination, initial fuel level, trip category (one-way and roundtrip), and vehicle range. The data from the AFDC website has many fields that describe the detail of the stations, such as the location, contact information, etc. The distance data generated by Mike Palmer includes only three attributes: source, target, and distance, and I will keep the same schema. The number of points will also affect the performance of the tool because it will require more computing power to run the tools. A potential significant change of the points will also change the data size of the pre-generated distance between all pairs of nodes.

**Data Manipulation**

This step combines the use of python and PHP programming, bootstrap, SQL, and AGOL geocoding. The PHP programming will deal with redesigning the user interface of the mapping tool, with the application of JavaScript API. Bootstrap also builds the front-end of the project, and it is a framework that can tweak the interface to be more responsive and mobile-friendly. The preprocessed networks generated by AGOL geocoding is the core of my work this project. The geocoding service has a python API that allows me to find the point via the “address” attribute in the csv data. Then I will be able to generate the shortest path between all pairs of stations using python scripting.

I switched to AGOL geocoding instead of the Google API because an ordinary user without a paid license to Google Maps API Premier is subject to a daily query limit of 2,500 geolocation requests and is likely to experience some ‘hiccups’ in executing the tool, although it does not need the preparation of a network dataset compared to other tools, such as ArcGIS network extension, (Wang, Xu, 2011). Google only gives 300 free trial credits for a regular user, and I need to figure out how much credits are needed for generating the distance between the nodes of all stations.

Figure 2 illustrates the workflow of Dr. Kuby’s group. My future work in this project will cover the first step (Raw street network, Google Maps API), and last step (Shortest Path Web GIS Display, Google Maps API). However, I will be using the AGOL geocoder instead of the Google API because ESRI will allow me to use my student user’s credits and this will also make it convenient for the future student developers.

A close up of a map

Description automatically generated

Figure 2. Flowchart of Solution Algorithm. (Kuby, 2014)

**Optional Work**

If there is extra time, I will use Neo4j to generate the shortest distance between the stations and compare its performance with PostgreSQL. The two technologies: relational databases and non- relational database, will remain in usage side by side, each with the perfect fit for its capabilities. Compared to RDBMS, Neo4J as a graph database management system which is considered as NoSQL, can query in real-time and may perform faster on some occasions. Because of its data structure based on key, value, nodes, and property, it is much easier to traverse the data than the RDBMS. Neo4J is widely used in real time fraud detection and building social network websites. It is used by companies such as Walmart, Ebay, Airbnb, HP, and Sisco. In some cases, Neo4j should be considered as an alternative for specific tasks. The comparison will show which database will perform faster in this particular case.

Other NoSQL database might be proper for the project, such as CouchDB and MongoDB, but their spatial features are limited. For example, CouDB is relatively slow in performance, and MongoDB only support points. On the other hand, Neo4j-Spatial has more spatial advantages that make it a potential competitor of PostGIS (Wang, Xu, 2011).

**Challenges**

The requests for generating the distance between the stations cost a large computational power. We could reduce the number of requests if we filter out the station pairs with Euclidean distance > 400 miles first, rather than requesting driving distances for all pairs first and then filtering out those > 400 miles. With this being considered, a pre-generated dataset of the distance of the nodes must exclude those which have more than 400-mile distance between them.

Understanding the previous project and explore new methods of data analysis are also very challenging. It is important to keep the front-end and back-end service tools consistent. It could cause potential problems if the artificial feasible network would be built from distances from 2 different sources. For instance, in the previous project, if Bing map calculated a route as 249 miles, and that segment and its distance are used in an optimal route by our tool, it is possible that when our tool generates the final output and map, it uses Google to calculate that segment, and if Google computes a different shortest-time route it could be 251 miles long for instance. If the user entered a driving range of 250, the input from the distance matrix would show 249 miles and the route would be feasible but the user would only see the output on the screen which could show 251 miles as the length of that leg of the trip and the user would think that our tool is not working properly. Using a consistent routing program throughout should help minimize these kinds of discrepancies that are caused by different sources of computing.

A close up of text on a white background

Description automatically generated

**Deliverables:**

|  |  |
| --- | --- |
| **Content** | **Memo** |
| Collection of current Updated data of the CNG and Hydrogen Station in 2020 | From the website <https://afdc.energy.gov> |
| Using AGOL geocoding service to pre-generate the shortest distance between all pairs of nodes of the stations. | .csv file |
| Python and PHP code, SQL query documents. | .py and .txt files |
| A repository on GitHub. | Potential future collaborative work |
| Web Application. | 1. Prototype for finding the shortest feasible route. http://gis.sparc.asu.edu/afvrouting/ 2. Tweak the web user interface to be responsive and mobile-friendly using Bootstrap. 3. Alter the web tool to also use the AGOL geocoder to make sure calculated network distances match up to the geocoder used for the start and end addresses, the user types in. |
| Report | In MLA format, with instructions and test results for each distance tool. |
| Presentation | August before graduation |

References

Baas, B. L. P. *Nosql spatial–neo4j versus PostGIS*. MS thesis. 2012.

Kuby, Michael, et al. "An efficient online mapping tool for finding the shortest feasible path for alternative-fuel vehicles." *International journal of hydrogen energy* 39.32 (2014): 18433-18439.

Melendez M. Transitioning to a hydrogen future: learning from the alternative fuels experience. Golden, CO: National Renewable Energy Laboratory; 2006. Technical Report No. NREL/TP-540e39423.

Wang, Fahui, and Yanqing Xu. "Estimating O–D travel time matrix by Google Maps API: implementation, advantages, and implications." *Annals of GIS* 17.4 (2011): 199-209.

Zandbergen, Paul A. *Python scripting for ArcGIS*. Esri press, 2015.