Geo Location Dataset Analysis for Identifying Fertilizer Dealers in Vellore district Tamilnadu, India

Rajkumar Rajasekaran1*, Thipireddy Rishith Reddy1, Palnati Gowtham Reddy1, Jolly Masih2

1School of Computer Science and Engineering, Vellore Institute of Technology, India
2Prestige Institute of Engineering Management and Research, India

*vitrajkumar@gmail.com

Abstract. The following data article is an insight to the data – Number of Dealers for Chemical Fertilizers in Vellore District as downloaded from the website of Mobile Fertilizer Management System [1] is maintained by Department of Fertilizers, Government of India. It can be downloaded in CSV format. The data gives us the knowledge about the dealer’s unique identification number, type of dealership, name of agency, address, license validity period, and other governmental proof ids. The file contains 579 records with certain missing values of which 472 records have address and dealership details specified. The data can be visualized as an aid to farmers so as to provide them fertilizers at a minimum cost by considering farmer’s location and the availability of type of dealers specified around them.

1. Introduction

The data provides information about number of dealers in Vellore district for both of the types wholesalers and retailers. Every record is given 2 additional attributes with respect to its location i.e. latitude and longitude. The attributes are obtained with the help of an efficient method which makes use of chromedriver.exe and the Selenium library from Python 3. The URL of Google Maps is used to obtain this location and its property of changing the URL on different queries of different addresses is what makes it useful here. The location of different dealers can be used to compare with farmer’s present location and provide him with the nearest dealer he must travel to so as to purchase the fertilizers at a cheaper travelling cost.

2. Methodology

A detailed discussion with company’s marketing team coupled with spatial and attribute data enabled us to simulate the market in GIS for identification of the most suited distribution Network. District and subdivision level digitized map with various layers such as National Highway, Main Road, other road, Rail, District headquarters, Main towns and other towns with subdivision level external database of Demography and agriculture like fertilizer consumption, Net area under cultivation, area with various main crops data are used in MapInfo package. The said company’s Distribution Network location has been created on a separate layer on the basis of their location. The analysis has been made with continuous interaction with marketing personnel to cover the ground reality from their experience. At the time of final selection the location of a town preference is judged on center of gravity basis and its accessibility to the local farmers. This Center of Gravity simulation and accessibility feature matched well in the MapInfo package.
3. Result and Discussion

3.1. Dataset

The original dataset is composed of details of 579 dealers. For each dealer its unique ID is mentioned. The dataset is ordered with respect to this identification number. For every dealer the record contains details about dealership type, his/her agency name, mobile number, Pan number, tin number, complete address, dealer license and license period. The data set is downloaded in CSV format from the specified website. It is then refined and cleaned before adding the location attributes of every dealer. The data in CSV format can be visualized as below.

![Figure 1. Comma-Separated Values format of dataset which includes unique ID of state, name of the state, name of the District, Number of wholesalers and the Number of retailers respectively](image)

3.2. Cleaning of Data

The dataset contained the rows corresponding to address values that were not specified. Such rows become irrelevant while computing the nearest district to farmer’s location as dealers for which address is not known cannot be located on the map. The data is cleaned by removing such irrelevant observations and stored again as CSV file corresponding to details of 472 dealers in different parts of Vellore.

3.3. Adding the Location Attribute to Data

For each row we will find the latitude and longitude of the dealer address specified. The latitude and longitude are found using the chromedriver.exe app. As we are using Python 3, we could make use of the selenium library which facilitates in the running the webdriver functions. The driver allows us to open a new Google Chrome tab. We put the google maps URL with the given address lines 1 and 2 as the query in the URL. The motivation behind using this approach is that the URL loaded onto the new window of the browser changes as the search is completely loaded. The new URL now contains the latitude and longitude value for the search query specified.

The initial URL looks like: https://www.google.com/maps/place/Address/ The address is searched as needed by placing it in the above URL. The new updated URL looks like: https://www.google.com/maps/place/37,+Thennamara+St,+Kosapet,+Vellore,+Tamil+Nadu+632001 The new URL contains the latitude and longitude value for the search query specified.

It can be visualized as shown in Figure 1 below. As we can see the text highlighted in the new URL is the co-ordinates of the place searched. The selenium library’s webdriver function now allows us to fetch this co-ordinates. This allows us to find the latitude and longitude for each dealer already present in the dataset. Pre-processing and storing these values will allow us to directly compute the output for the user specified query. The processed data will be JSON file having a list of dictionary objects. Each dictionary object corresponds to one of the districts present in the dataset.
Figure 2. The red rectangle highlights the latitude and longitude values after the query URL is loaded completely. Also the red marker on the map indicates the location of the search query.

Description of JSON Data

The JavaScript Object Notation is a representation of data in terms of a collection of number of key-value pairs. This collection is known as a dictionary in Python. The JSON file formed is the list of such dictionaries, each corresponding to a dealer in the dataset. Each dictionary can be represented as the following table.

Table 1. The description of each dictionary data structure in the JSON file corresponding to every dealer in the dataset

| Keys          | Data Types     | Description                                      |
|---------------|----------------|--------------------------------------------------|
| Dealer ID     | Integer        | The unique ID value corresponding to a dealer.   |
| Dealer Type   | String         | The type of dealership – Retailer or Wholesaler. |
| Agency Name   | String         | The name of the agency.                          |
| Mobile Number | Long Integer   | Dealer’s mobile number.                          |
| PAN number    | String or NaN  | The PAN card details of the dealer if available. |
| Tin number    | String or NaN  | The Tin number of the dealer if available.       |
| Address Line 1| String         | The first address line of dealer.                |
| Address Line 2| String         | The second address line of the dealer.           |
| License Issue | Date           | The issuing date of the license.                 |
| License Expiry| Date           | The expiry date of the license.                  |
| Latitude      | Floating Point | Latitude of district.                            |
| Longitude     | Floating Point | Longitude of district.                           |

4. Conclusion

The dataset in the JSON format is useful in comparing the farmer’s present location with every set of latitude and longitude corresponding to a dealer. The dataset is available for every state in India and all its different districts. The farmer’s present location can be obtained using GPS or using a method similar to one described above. Various APIs are available to compute the distance between 2 coordinates on the map or otherwise we can also make use of Haversine formula which computes distance between a pair latitudes and longitudes. Thus, the Geo-location dataset is useful in providing farmers the chemical fertilizers at a comparatively cheaper cost of travelling.

References

[1] http://mfms.nic.in/
[2] http://164.100.128.10/mfmsReports/getDealerInfoExcel
[3] Gojare, Satish & Joshi, Rahul & Gaigaware, Dhanashree. (2015). Analysis and Design of Selenium WebDriver Automation Testing Framework. Procedia Computer Science. 50. 341-346. 10.1016/j.procs.2015.04.038.

[4] Devi, Jyoti & Bhatia, Kirti & Sharma, Rohini. (2017). A Study on Functioning of Selenium Automation Testing Structure. International Journal of Advanced Research in Computer Science and Software Engineering. 855-862. 10.23956/ijarcsse/V7I5/0204.