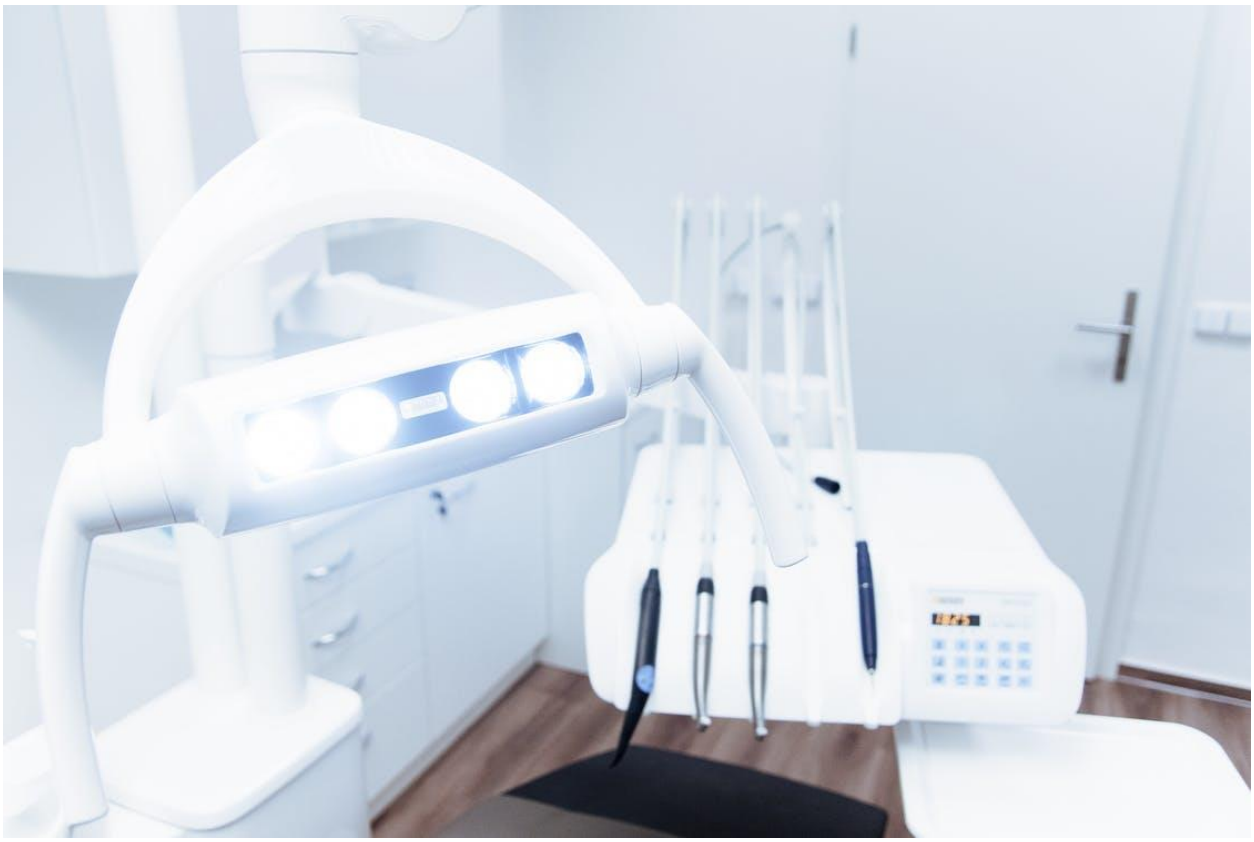


Clinic Go Where: KMeans-Based Clinic Recommendation System

Document: Installation and User Guide

Last Updated: 02-May-2020

Disclaimer: This project is a school project. The author does not own the data used in this project, please do not use the data for commercial projects.

Contents

1. Prepare Python Environment

1.1. You are recommended to install Anaconda for this configuration
1.2. From window search, start Anaconda Prompt
1.3. Create a new virtual environment for the project from Anaconda Prompt by typing “conda create --name rpa”
1.4. Activate the environment by typing “activate rpa”
1.5. Install the requirements by typing “pip install -r requirements.txt”
1.6. From window search, start Anaconda Navigator
1.7. Select “RPA” environment & start editors under this environment

2. Set Up Google API

2.1. Log in google cloud platform (GCP) https://console.cloud.google.com/
2.2. Set up google account based on GCP instructions
2.3. From google search bar, find “Geolocation API” and enable it
2.4. From google search bar, find “Distance Matrix API” and enable it
2.5. From google search bar, find “Direction API” and enable it
2.6. From GCP home, click “API and Services” (left hand navigation bar)
2.7. Click “Credentials”
2.8. Click “Create Credentials”
2.9. GCP will generate a key. Please replace “Your Google API” with the key in these files: scripts\ Model.ipynb and app\app.py. Example:

```
google_maps = googlemaps.Client(key='Your Google API')
```

3. Start Web App

3.1. Start Spyder(recommended)/ Pycharm (have to re-load the environment, please see step 7 for more info) /any other editor from Anaconda. Make sure the environment is properly configured & loaded
3.2. Run app/app.py. Access the web app from http://127.0.0.1:5000/
3.3. Enter a Singapore postal code
3.4. The web app will return you a list of nearby clinics and their details (please refer to step 6 for sample use case)

4. Download Clinics Data

4.1. You can download clinics data from scripts\ ExtractData _HCI.ipynb
4.2. You can download phpc clincis data from scripts\ExtractData_PHPC.ipynb
4.3. You can download Singapore geolocation from scripts\ Extract_PostalCodes.ipynb
4.4. Or you can use the pre-downloaded data under scripts\data folder to run the app. Please remember to unzip address_list.7z (I was not unable to upload the original file due to size issue.)

5. Process Data

5.1. Please run scripts\ ProcessData.ipynb to clean the data downloaded

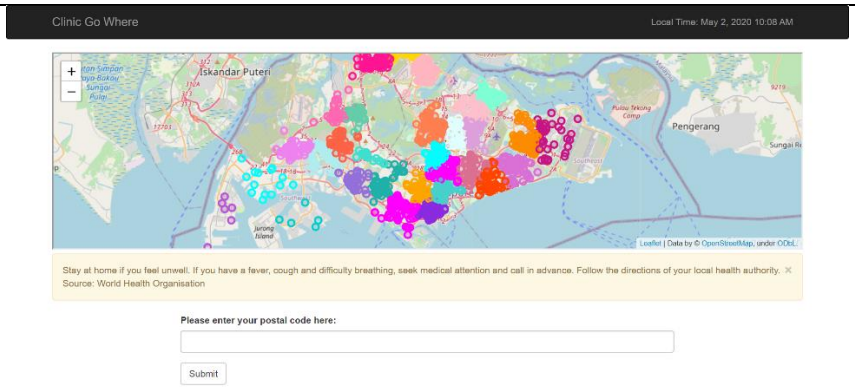
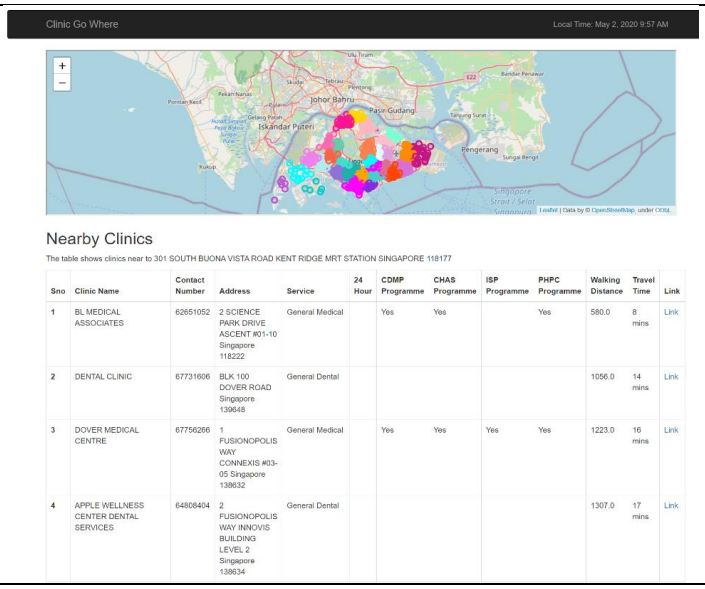
5.2. Please run scripts\ Model.ipynb to generate the clustering models

5.3. Or you can use pre-processed data under app\data folder to run the webapp

6. Sample Use Case

6.1. Enter a valid postal code. If invalid codes are entered, error prompts will appear:

6.2. Enter a valid code like “118177”. The app will try to recommend top 10 nearest clinics with details.

	Screenshot
Search Page	
Results Page	

Sno	Clinic Name	Contact Number	Address	Service	24 Hour	CDMP Programme	CHAS Programme	ISP Programme	PHPC Programme	Walking Distance	Travel Time	Link
1	BL MEDICAL ASSOCIATES	62651052	2 SCIENCE PARK DRIVE ASCENT #01-10 Singapore 118222	General Medical		Yes	Yes		Yes	580.0	8 mins	Link
2	DENTAL CLINIC	67731606	BLK 100 DOVER ROAD Singapore 139648	General Dental						1056.0	14 mins	Link
3	DOVER MEDICAL CENTRE	67756266	1 FUSIONOPOLIS WAY CONNEXIS #03-05 Singapore 138632	General Medical		Yes	Yes	Yes	Yes	1223.0	16 mins	Link
4	APPLE WELLNESS CENTER DENTAL SERVICES	64808404	2 FUSIONOPOLIS WAY INNOVIS BUILDING LEVEL 2 Singapore 138634	General Dental						1307.0	17 mins	Link

7. Load Project Environment in Pycharm

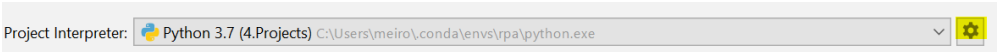
7.1. Follow step 1 to activate “RPA” environment

7.2. Find the path of Anaconda virtual environment by typing “echo %CONDA_PREFIX%” for window user (for MAC/Linux user, please google the command)

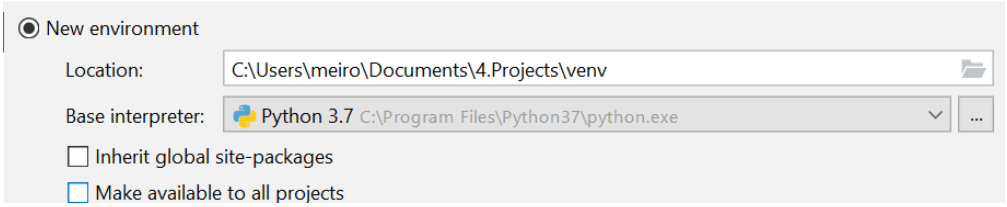
7.3. Start Pycharm

7.4. From File >> Open to load the project

7.5. From File >> Settings >>Project Interpreter to load the environment (by clicking the icon highlighted in yellow)



7.6. Select Add…>>Virtual Environment >> New environment >> Base interpreter >> Select the python.exe under the path shown in step 7.2



7.7. Save the settings

8. System Spec

8.1. Python Flask Framework is used for web app development

8.2. Tagui is a Robotic Process Automation (RPA) tool used for data collection

8.3. The machine learnings models are mostly trained using sklearn libraries

8.4. OneMap API Singapore and Google API are used for location and distance search

8.5. No DB is used in this project. Data are stored in JSON/CSV format

9. Special Notes

9.1. Please feel free to contact me at e0384977@u.nus.edu if more clarification is needed.