LEVERAGING OPEN API: INTERCHANGE NETWORK DATA FOR ACCESS TO HEALTH SERVICES IN URBAN AREAS

Ahmad Najmi Ariffin[1,2], Jamaliah Jaafar[1], Shahida Mohd Said[2],
Zainal Abidin Abd Mutalib[3]

ABSTRAK

Kertas kajian ini, menyampaikan pendekatan yang memanfaatkan teknologi antara muka pengaturcaraan aplikasi (API) untuk menilai akses bandar ke perkhidmatan kesihatan dan memetakan akses yang sedia ada. Dengan menggunakan pakej Pandana Python, struktur data diwujudkan untuk setiap rangkaian dengan hanya menggunakan beberapa baris kod. Permintaan data rangkaian dari OpenStreetMap (OSM), menghasilkan data perkhidmatan kesihatan berdasarkan data log masuk pengguna OSM dan mengira pelbagai ukuran akses ke destinasi fasiliti kesihatan. Walaupun terdapat kelebihan pendekatan sistem maklumat geografi (GIS), kami berpendapat capaian rangkaian adalah lebih tepat bagi menggambarkan bagaimana orang ramai berinteraksi dengan persekitaran (Maurer, 2020). Permintaan data rangkaian dalam lingkungan kawasan bounding box di Kuala Lumpur, Malaysia menghasilkan data OSM dengan 83,799 nod dan 115,874 edge table. Ia hanya memerlukan 206.11 saat untuk memuat turun data OSM 36,231.2 kilobyte. Perlombongan data masa-nyata mendapatii sebanyak 74 klinik pergigian, 2 hospital, 267 klinik, 140 stesen pengangkutan awam dan 23 platform pengangkutan awam terletak di kawasan carian. Parametre yang ditetapkan adalah pada jarak maksimum 1000m hingga 3 POI terdekat dengan jarak klinik pergigian dianggarkan dalam purata jarak 962.29m (±137.89). Pengiraan agregat menunjukkan bahawa 6 unit klinik pergigian terletak dalam jarak 500m disetiap nod dan 10 unit dalam jarak 1000m. Taburan spatial perkhidmatan kesihatan di Kuala Lumpur membuktikan bahawa akses ke beberapa tempat di seluruh bandar adalah dalam masa perjalanan kurang dari 2 minit dari satu lokasi ke lokasi lain. Pelaksanaan API akan mengurangkan waktu pembangunan, memastikan kualiti data yang tepat, dan memperluaskan ketersediaan pelbagai aplikasi pengurusan kesihatan awam.

Kata kunci: Analisis ruang; API; Bahasa pengaturcaraan Python; Perkhidmatan kesihatan

---

[1,2] Ahmad Najmi Ariffin is currently Assistant Director of Core Team Big Data Analytics (CTADR), Department of Statistics Malaysia and Postgraduate Student in the Centre for Restorative Dentistry, Faculty of Dentistry, The National University of Malaysia (UKM); [1] Jamaliah Jaafar is currently Head of Team in Core Team Big Data Analytics, Department of Statistics Malaysia; [2] Shahida Mohd Said is currently is Associate Professor in Department of Restorative Dentistry, Faculty of Dentistry, The National University of Malaysia (UKM); and [3] Zainal Abidin Abd Mutalib is currently Assistant Director of the Population and Demographic Statistics Division, Department of Statistics Malaysia.

ABSTRACT

In this paper, we present an approach that leverages on application programming interface (API) technologies to evaluate accessibility to health services in urban areas and mapping the existing access to health services. Using Pandana Python package, the underlying data structures is created for each network with only a few lines of code. The queries seize a network data from OpenStreetMap (OSM), generate the health services data that users of OSM have logged, and calculated values of various measures of access to health facility destinations. Although there are advantages to the geographic information system (GIS) approach, we think network queries are more accurately depicting how people interact with environment (Maurer, 2020). Requesting network data within bounding box in Kuala Lumpur, Malaysia returning OSM data with 83,799 nodes and 115,874 edge table creation. It took 206.11 seconds to complete download 36,231.2 kilobytes OSM data. Real-time mining found 74 dental clinics, 2 hospitals, 267 clinics, 140 public transport stations and 23 public transport platforms located within designated area. Parametres are set at a maximum distance of 1000m for up to the 3 nearest POIs: dental clinics, the proximity distances calculated in average 962.29m (±137.89). Calculation of an aggregation showed that 6 units dental clinics are located within 500m of each node and 10 units within 1000m. Spatial distribution of health services within Kuala Lumpur proved that citywide accessibility across multiple places is less than 2 minutes travel time from one location to another. Implementation of API will reduce development time, ensuring data quality accuracy, and expanding the availability of various public health management applications.

Keywords: Spatial analysis; API; Python Programming Language; Health services

1. INTRODUCTION

In this paper, we present an approach that takes advantage of these API technologies (a) to compute shortest paths between the locations of dental clinic and hospital, (b) simplify measurement of dental clinic proximity to each network node, (c) evaluate access to health services in urban areas, and (d) mapping and identify the challenges of access to dental health services. In Australia, the study of distance by line-of-the-site has been used as a close proxy. (McGuire, Kruger, & Tennant, 2011). In Malaysia, the need arose to develop a basic model for the country's existing public oral health services using GIS to identify access gaps. Dental services were distributed for high population density and were unevenly distributed throughout Malaysia. (Md Bohari, Kruger, John & Tennant, 2019). The use of census data and GIS tools to target health services in areas of greatest need is not a new concept in health care. Recent studies on health care using GIS used travel time as a measure of access. This is of added significance to the preparation of formulating a national oral health policy for Malaysia. The development of web-based, open source-based health care facility sites with precise street addresses are imperative for many developing countries such as Malaysia where such information systems are underdeveloped.

2. LITERATURE REVIEW

The use of GIS allows social and health related patterns to be studied within their spatial contexts, as well as offering a more easily understandable way of communicating spatial data. The use of mapping software, for example, enables spatial data to be presented in a much more consumable manner than tables of data or text (Broomhead, Ballas, & Baker, 2019). It may also be important to validate this basic geographic model of public dental care distribution with the latest population data for GIS applications.

An API built into network queries provides a seamless and more dynamic user experience. Although there are benefits to the GIS approach, (Sam Maurer, 2020) believe that the network queries are a more accurate representation of how people interact with their surroundings. Spatial approach provides an innovative decision-support tool in health informatics with great potential in planning, monitoring and evaluation of public health services.

3. RESEARCH METHODS

In the application of using Pandana extension (Python package), the underlying data structures are created for each network. With only a few lines of code, the queries seize a network data from OpenStreetMap (OSM), generate the health services information that users of OSM have recorded, and calculated values of various measures of access to destinations on the street network. This requires installing a Pandana extension called OSMnet.

- conda install osmnet or pip install osmnet.

As we decided, the designated boundary within Kuala Lumpur city centre have been selected for this session. The exact boundary area are determine using free tools at http://boundingbox.klokantech.com/ to get the coordinates of bounding boxes as shows in Figure 3.1.

```
network = osm.pdna_network_from_bbox(3.033554, 101.615081, 3.244325, 101.7592)  # KL, M
alaysia
```

```
Requesting network data within bounding box from Overpass API in 1 request
(s)
Posting to http://www.overpass-api.de/api/interpreter with timeout=180, "
{'data': '[out:json][timeout:180];(way["highway"]["highway"!~"motor|propos
ed|construction|abandoned|platform|raceway"]["foot"!~"no"]["pedestrians"!
~"no"](3.03355400,101.61508100,3.24432500,101.75920000);>;);out;'}"
Downloaded 36,231.2KB from www.overpass-api.de in 35.16 seconds
Downloaded OSM network data within bounding box from Overpass API in 1 req
uest(s) and 37.01 seconds
Returning OSM data with 240,653 nodes and 66,270 ways...
Edge node pairs completed. Took 159.11 seconds
Returning processed graph with 83,799 nodes and 115,874 edges...
Completed OSM data download and Pandana node and edge table creation in 20
6.11 seconds
```

Figure 3.1: The Exact Boundary Area

3.1 Saving and reloading networks

The Pandana network object/ data are save directly by easily recreating it from the nodes and edges as shown in Figure 3.2.

In [8]:

```
network.nodes_df.to_csv('nodes.csv')
network.edges_df.to_csv('edges.csv')
```

In [9]:

```
nodes = pd.read_csv('nodes.csv', index_col=0)
edges = pd.read_csv('edges.csv', index_col=[0,1])
```

In [10]:

```
edges
```

Out[10]:

		from	to	distance
28918006	**1544290930**	28918006	1544290930	52.34
	1814470452	28918006	1814470452	29.38
28918008	**2084225785**	28918008	2084225785	11.02
28918012	**6585754593**	28918012	6585754593	59.68
28918021	**4058187858**	28918021	4058187858	38.00
...	**...**	...	...	...
7822849676	**7434567185**	7822849676	7434567185	10.49
7822870262	**2686197735**	7822870262	2686197735	34.09
7822870274	**2168760653**	7822870274	2168760653	6.31
	7822849673	7822870274	7822849673	75.65
7824819982	**5789205670**	7824819982	5789205670	46.25

117438 rows × 3 columns

Figure 3.2: Recreate it from the nodes and edges

The network comprised of a set of nodes and edges as Data Frames (Figure 3.3).

```
network = pandana.Network(nodes['x'], nodes['y'],
                          edges['from'], edges['to'], edges[['distance']])
```

Figure 3.3: Nodes and edges as Data Frame

There are a few general steps in measuring a variable using network queries. First start with creating and preprocessing the network, then assign a variable to the network, and finally perform the aggregation.

3.2 Load the health service locations data

Here, the locations of health services are loaded and listed from Open Street Map (using the same OSMnet extension as below in Figure 3.4), and then calculate some shortest paths between the locations.

In [12]:

```
dental = osm.node_query(
    3.033554, 101.615081, 3.244325, 101.7592, tags='"amenity"="dentist"')
```

In [15]:

```
hospital = osm.node_query(
    3.033554, 101.615081, 3.244325, 101.7592, tags='"amenity"="hospital"')
```

In [16]:

```
clinic = osm.node_query(
    3.033554, 101.615081, 3.244325, 101.7592, tags='"amenity"="clinic"')
```

In [17]:

```
pubt = osm.node_query(
    3.033554, 101.615081, 3.244325, 101.7592, tags='"public_transport"="station"')
```

In [18]:

```
pubstop = osm.node_query(
    3.033554, 101.615081, 3.244325, 101.7592, tags='"public_transport"="stop_position"'
)
```

Figure 3.4: OSM Query

3.3 Statistical analyses

All data for analysis were extracted using Pandana extension called OSMnet package from Overpass API, the integrated database in Open Street Map into Jupyter Notebook. Descriptive statistical analyses (including number of dental clinics, hospitals, public transport stations and platform within designated boundary), calculation of shortest paths, proximity, accessibility and visualization were completed using Python programming language.

Parametres are set at a maximum distance of 1000 metres for up to the 3 nearest points of interest (POIs). Pandana aggregation, showed that 6 units dental clinic are located within 500 metres of each node and 10 units dental clinic are within 100 metres.

4. RESULTS

Once the calculation has been done, the result of calculations can be displayed with variety visualization techniques available. Requesting network data within bounding box (Kuala Lumpur, Malaysia: 3.033554, 101.615081, 3.244325, 101.7592), returning OSM data with 240,653 nodes and 66,270 ways. It took 206.11 seconds to completed OSM data download (36,231.2KB) files size; Pandana 83,799 nodes and 115,874 edge table creation.

In [21]:

```
dental
```

Out[21]:

id	lat	lon	addr:city	addr:place	amenity	healthcare	name	name:en	adc
1538397872	3.09	101.65	Petaling Jaya	PJ Old Town	dentist	dentist	Klink Pergigian Chong	Chong Dental Surgery	
1583831995	3.10	101.65	NaN	NaN	dentist	dentist	Pristine Dental Centre	NaN	
1858243858	3.19	101.70	NaN	NaN	dentist	dentist	Klinik Pergigian Yong	NaN	
2355670006	3.17	101.70	NaN	NaN	dentist	NaN	Jabatan Pergigian	NaN	
2914942039	3.15	101.67	NaN	NaN	dentist	dentist	Dentalpro	NaN	
...	...	...	...	...	...	...	...	...	
7368450050	3.14	101.63	NaN	NaN	dentist	dentist	Prema Nathan Dental Surgery	NaN	
7391369327	3.14	101.63	NaN	NaN	dentist	dentist	Chan Yeo Ho	NaN	
7491178583	3.16	101.70	NaN	NaN	dentist	dentist	Klinik Pergigian Fauziah	NaN	
7513290698	3.15	101.65	NaN	NaN	dentist	dentist	iCare Dental	NaN	
7625135311	3.14	101.73	NaN	NaN	dentist	NaN	NaN	NaN	

74 rows × 25 columns

Figure 4.1: OSM Query

4.1 Shortest paths

4.1.1 Shortest route between facilities

The nearest node to each dental clinic/ hospital/ public transportation as identified by the system are shown in Figure 4.2.

In [29]:

```
network.shortest_path_length(nodesd[0], nodesd[1])  #dental
```

Out[29]:

```
1883.398
```

Figure 4.2: Shortest Path Query

The unit used for network's distance is in metre. The distance is between the locations of each dental clinic and between locations of each hospital. Pandana can generate several million shortest paths in less than a minute. The approximately shortest path between the locations of dental clinic is 1883 metres and 1552 metres between hospital locations.

4.2 Proximity

There are some procedures to find the nearest dental clinic/ hospital to each node. To make a network-wide calculation like this, we first need to formally add the dental clinic and hospital to the network. The *maxdist* and *maxitems* parametres are the maximum distance and item you count using proximity searches, so that Pandana can optimize the caching.

In [40]:

```
results = network.nearest_pois(distance = 1000,
                               category = 'dentist',
                               num_pois = 3,
                               include_poi_ids = True)
results.head()
```

Out[40]:

	1	2	3	poi1	poi2	poi3
id						
28918006	594.61	674.97	692.58	7242330986.00	7242331187.00	7242307489.00
28918008	304.69	370.33	503.41	7242331187.00	7242330986.00	7242308387.00
28918012	86.54	659.45	708.54	7242307489.00	7242307487.00	7242330986.00
28918021	376.38	453.42	463.48	7242331187.00	7242330985.00	7242308387.00
28930674	102.06	301.77	346.27	7242331185.00	7242307488.00	7242307487.00

Figure 4.3: Shortest Path Query for 1000 metres

These are the distances (in metres) and IDs of the three closest dental clinics to each network node.

In [45]:

```
network.set_pois(category = 'hospital',
                 maxdist = 5000,
                 maxitems = 3,
                 x_col = hos.lon,
                 y_col = hos.lat)
```

In [46]:

```
results2 = network.nearest_pois(distance = 5000,
                                category = 'hospital',
                                num_pois = 3,
                                include_poi_ids = True)
results2.head()
```

Out[46]:

id	1	2	3	poi1	poi2	poi3
28918006	4015.93	5000.00	5000.00	5993583360.00	nan	nan
28918008	4013.58	5000.00	5000.00	5993583360.00	nan	nan
28918012	3469.86	5000.00	5000.00	5993583360.00	nan	nan
28918021	4368.66	5000.00	5000.00	5993583360.00	nan	nan
28930674	2679.88	5000.00	5000.00	5993583360.00	nan	nan

Figure 4.4: Shortest Path Query for 5000 metres

Proximity results, the distances and IDs of the 3 closest dental clinics to each network node are 962.29 metre (± 137.89), 913.41 metre (± 208.95) and 975.34 metre (± 107.70), respectively.

4.3 Accessibility

How many dental clinics and hospitals are within 1000 metres of each node? Pandana calls this kind of calculation an aggregation. It requires passing a list of network nodes and associated values. In this case, the ‘value’ are dental clinics and hospital, but it could also be characteristics like square footage of a building or income. The computation means, amounts, percentiles, and other metrics, as well as applying linear or exponential ‘decay’ to more distant values are shown in Figure 4.5.

```
In [47]:
dental_nodes = network.get_node_ids(dental.lon, dental.lat)

In [48]:
network.set(dental_nodes,
            name = 'dentist')

In [55]:
accessibility = network.aggregate(distance = 1000,
                                  type = 'count',
                                  name = 'dentist')
accessibility.describe()

Out[55]:
count   83799.00
mean        0.50
std         1.30
min         0.00
25%         0.00
50%         0.00
75%         0.00
max        10.00
dtype: float64
```

Figure 4.5: Dental Clinics within 1000 metres

4.4 Visualization

4.4.1 Mapping dental clinic accessibility

Real-time mining indirectly found that 74 dental clinics, 2 hospitals, 267 clinics, 140 public transport stations and 23 public transport platforms are located within designated area as shown in Figure 4.7 and 4.8.

```
In [59]:
import matplotlib
from matplotlib import pyplot as plt

print(matplotlib.__version__)

3.3.1
```

```
In [125]:
fig, ax = plt.subplots(figsize=(20,18))

plt.title('KL: Dental Clinic within 500m')
plt.scatter(network.nodes_df.x, network.nodes_df.y,
            c=accessibility, s=12, cmap='YlOrRd',
            norm=matplotlib.colors.LogNorm())
cb = plt.colorbar()

plt.show()
```

```
In [172]:
fig, ax = plt.subplots(figsize = (15,14))
ax.scatter(dc.longitude, dc.latitude, c='blue', s=20)
ax.scatter(clinic.longitude, clinic.latitude, c='purple', s=15)
#ax.scatter(pubt.longitude, pubt.latitude, alpha=0.2, c='black', s=15)
#ax.scatter(pubstop.longitude, pubstop.latitude, c='y', s=15)
ax.scatter(hos.longitude, hos.latitude, c='r', s=150)
ax.set_title('Plotting Spatial Data of Dental Clinics in Kuala Lumpur')
ax.set_xlim(BBox[0],BBox[1])
ax.set_ylim(BBox[2],BBox[3])
ax.imshow(ruh_m, zorder=0, extent = BBox, aspect= 'auto')

Out[172]:
<matplotlib.image.AxesImage at 0x25d37d8cb80>
```

Figure 4.6: Plot Query

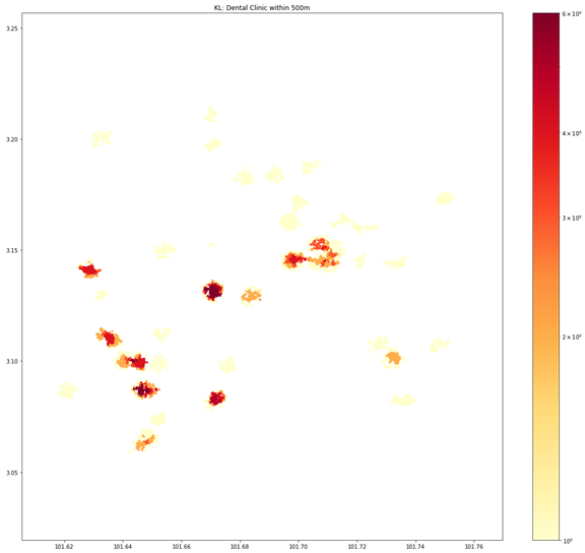

Figure 4.7: Dental Clinics in Kuala Lumpur within 500 metres

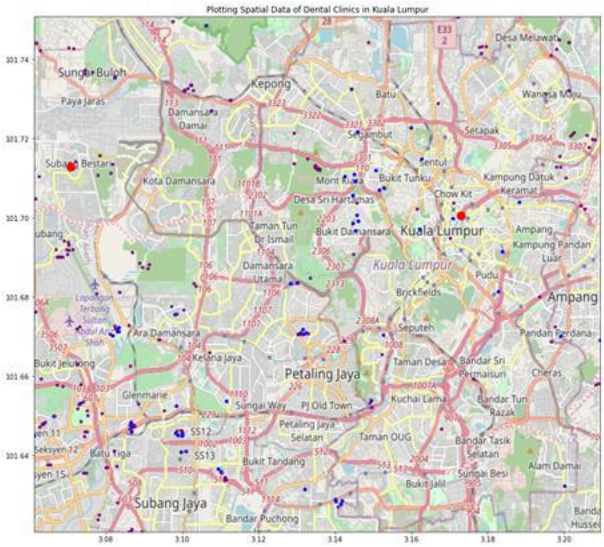

Figure 4.8: Plotting Spatial Data of Dental Clinics in Kuala Lumpur

5. CONCLUSION AND RECOMMENDATIONS

A snapshot of the findings shows that aggregate information for each source node in the network is the primary value of this study. Spatial distribution of health services locations within Kuala Lumpur proved that city-wide accessibility across multiple cities is less than 2 minutes travel time from one location to another health service. The best solution is an API, interface with pieces of code that allow applications to communicate with each other. It will reduce development time, improve accuracy and effectiveness of data quality, and the availability of various public health management applications. Applying disadvantage-based mapping and service targeting access to safety net care can be greatly enhanced and efficiency gains made (Crawshaw, Bunton & Gillen, 2003). This study has critical outcomes for the planning of future services in developing a network model for dental care.

Further analysis of the effects of variables such as relative household wealth in the public's approach to access dental care needs to be done. However, based on the data presented in this study, health planners may adopt a more evidence-based approach to planning new service network structures. Therefore, this study highlights the continuing need to monitor the geographical distribution of services to match the evolving socio-economic scenario of dental disease. The spatial approach provides a useful decision support tool in health informatics with a substantial potential to harness in planning public health care services (Perera, Kruger & Tennant, 2012).

REFERENCES

Bohari, N. F., Kruger, E., John, J., & Tennant, M. (2019). Analysis of dental services distribution in Malaysia: A geographic information systems – based approach. *International Dental Journal, 69*(3), 223-229. doi:10.1111/idj.12454

Bounding Box Tool: Metadata Enrichment for Catalogue Records by Visually Selecting Geographic Coordinates (Latitude/Longitude) for Maps. (n.d.). Retrieved from https://boundingbox.klokantech.com/

Broomhead, T., Ballas, D., & Baker, S. R. (2018). Application of geographic information systems and simulation modelling to dental public health: Where next? *Community Dentistry and Oral Epidemiology*. doi:10.1111/cdoe.12437

Crawshaw, P., Bunton, R., & Gillen, K. (2003). Health Action Zones and the problem of community. *Health & Social Care in the Community, 11*(1), 36-44. doi:10.1046/j.1365-2524.2003.00397.x

F.F. (2020a, August 5). Pandana — Pandana 0.5.1 documentation. Pandana v0.5.1. Retrieved from http://udst.github.io/pandana/

Mcguire, S., Kruger, E., & Tennant, M. (2011). Travel patterns for government emergency dental care in Australia: A new approach using GIS tools. *Australian Dental Journal, 56*(4), 389-393. doi:10.1111/j.1834-7819.2011.01371.

Perera, I., Kruger, E., & Tennant, M. (2012). GIS as a Decision Support Tool in Health Informatics: Spatial Analysis of Public Dental Care Services in Sri Lanka. *Journal of Health Informatics in Developing Countries, 6*(1). Retrieved from https://www.jhidc.org/index.php/jhidc/article/view/82

S.M.M.A.U.R.E.R. (2020b, August 6). UDST/pandana: Pandas Network Analysis: Fast accessibility metrics and shortest paths, using contraction hierarchies. Retrieved from https://github.com/UDST/pandana/