Route Subnetwork Generation using OpenStreetMap Data for Emergency Response Problem Modelling in Indonesia

Yohanes Gultom
Faculty of Computer Science
Universitas Indonesia
Depok, Indonesia
yohanes.gultom@ui.ac.id

Heru Suhartanto
Faculty of Computer Science
Universitas Indonesia
Depok, Indonesia
heru@cs.ui.ac.id

***Abstract*—Route subnetwork generation is useful in modelling disaster emergency response operation problems such as evacuation, aid distribution, and personnel scheduling. Through this paper, we propose an end-to-end approach of generating subnetwork based on list of point of interests (villages, shelters, depots, etc) and publicly available data using combination of opensources tools. The end result is an opensource software available in public code repository. We also present some experiment results for three areas in Indonesia: Jakarta, Lombok, and Yogyakarta.**

***Index Terms*—subnetwork, openstreetmap, emergency response**

I. Introduction

Route subnetwork generation is a process of extracting some portion of map data (eg. OpenStreetMap data) to build a graph where point of interest (POI) nodes are connected by minimum required routes as edges. This subnetwork is useful in modelling emergency response problems such as evacuation, aid distribution, and personnel scheduling.

The goal of the subnetwork generation is to process list of POI (POI ID, POI category, latitude, and longitude) to generate a subnetwork network/graph where:

1) Each nodes represents POI or route intersection
2) Each edges represents route between node
3) Only routes that are required to connect POIs included
4) Each nodes has risk index (greater value means greater risk)
5) Each routes has risk index derived from its nodes' risk

The example expected subnetwork is visualized in Figure 1.

II. Related Works

OpenStreetMap.org (OSM) [1] is an open community-driven map data provider that provides standard and data needed to build a subnetwork for emergency response problem modellling. The problem is that it contains too much data so filtering and extraction are needed.

OpenStreetMap.id is a website that provides extracted OSM data for most provinces in Indonesia in Protocolbuffer Binary Format (PBF). This website solved the issue regarding regional data extraction.

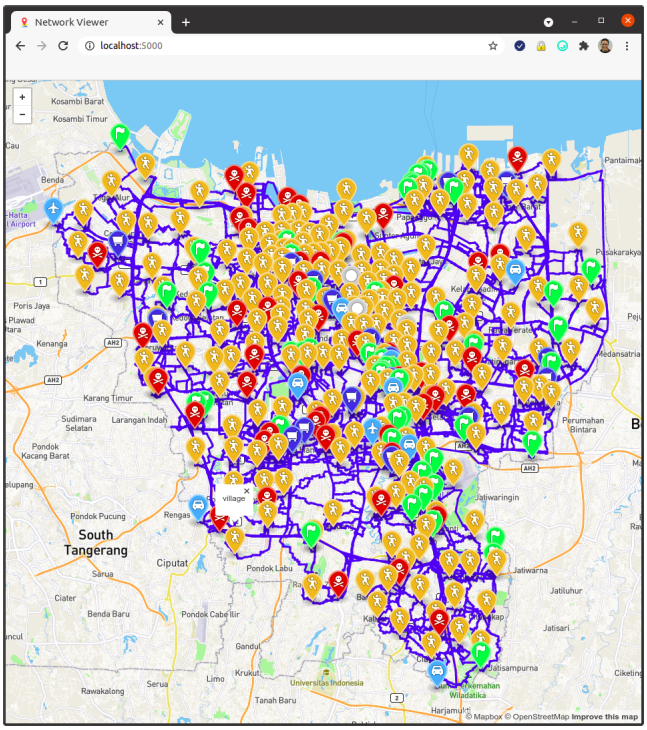

Fig. 1. Visualization of Jakarta's subnetwork during flood disaster

INARISK[1] is a public Geographic Information System provided by The National Agency for Disaster Countermeasure of Indonesia. This system provides information about the disaster risk index of all regions in Indonesia in a form of colored map layer as shown in Figure 2.

OsmToRoadGraph[2] is an opensource tool to generate graph/network from raw OSM data. Since the OpenStreetMap.id only provides PBF file, we will need additional

[1]http://service1.inarisk.bnpb.go.id:6080/arcgis/rest/services/inaRISK
[2]https://github.com/AndGem/OsmToRoadGraph

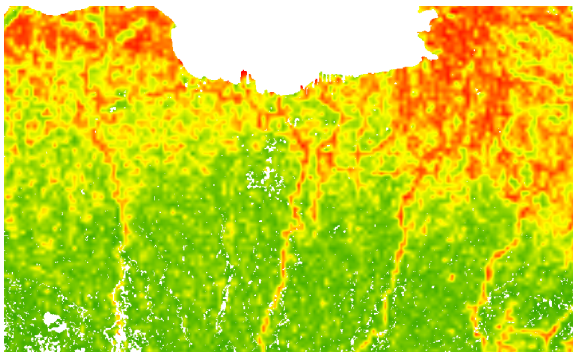

Fig. 2. Jakarta flood disaster risk index from INARISK

tool called Osmconvert[3] to convert PBF to OSM before using OsmToRoadGraph to generate graph/network data in PYCGR/PYCGRC and JSON format.

PostGIS[4] is an spatial data extension of one of the most popular opensource database, PostgreSQL[5]. PostGIS allows us to map each POIs provided by user to their nearest node in PYCGR file. The only thing left to solve is to remove unecessary nodes and routes from the graph to reduce the size of final input data for the model.

NetworkX[6] is an opensource Python package that provides data structure and APIs for complex graph search and manipulations. This package provides a way to remove unecessary nodes and routes in order to generate minimized subnetwork/subgraph suitable for modelling.

III. Methodology

In a nutshell, the methodology that we propose is combining related tools into a new system to generate route subnetwork from POIs of emergency response operation. The flowchart of the system is shown in Figure 3.

These are the steps that our proposed subnetwork generator does to generate the route subnetwork:

1) Loads OSM nodes information from PYCGR/PYCGRC file to PostGIS-extended PostgreSQL database
2) Loads POI list and find nearest OSM nodes
3) Loads JSON graph and finds minimal nodes/edges required by combining shortest paths of POIs permutation pairs
4) Finds risk index of each nodes by converting pixel colors in INARISK risk layer image. Manually searched pixel-to-coordinates samples are required to estimate the corresponding pixel for each node's coordinates (latitude/longitude)
5) Writes output files that represent the route subnetwork, risk index, and mapping to input POIs

Additionally we also provide a simple webserver that render the subnetwork on top of interactive map as shown in Figure 1.

[3]https://wiki.openstreetmap.org/wiki/Osmconvert
[4]https://postgis.net/
[5]https://postgresql.org/
[6]https://networkx.org/

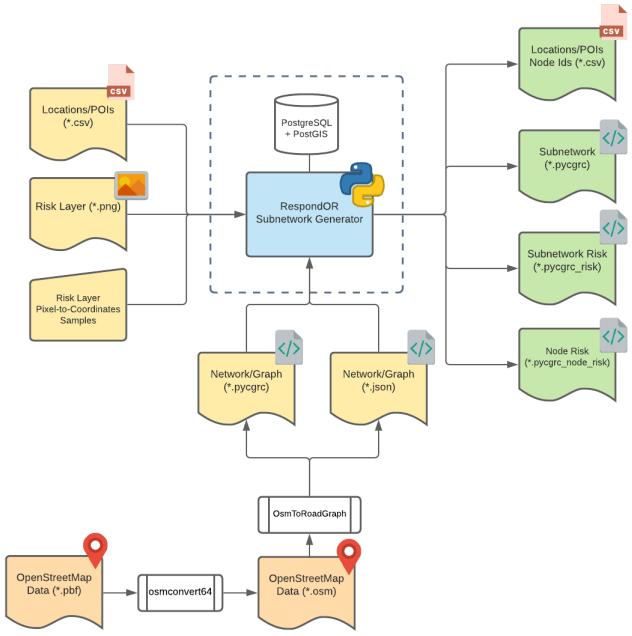

Fig. 3. End-to-end network generation system flowchart

The source code of our proposed system along with its usage guide are currently published in a publicly accessible code repository[7].

IV. Experiments

In order to validate and test performance of our proposed system, we generated route subnetwork for three emergency response operations:

1) Yogyakarta earthquake
2) Jakarta flood
3) Lombok earthquake

We used a computer with following specifications used to run the experiments:

- CPU: AMD Ryzen Threadripper 3970X 32-Core Processor
- RAM: DDR4 64 GB
- Storage: SCSI 4 TB
- OS: Ubuntu 18.04

The experiment results are shown in Table I.

TABLE I
Experiment Results

No	Operation	POI	Nodes	Routes	Processing Time
1	Yogyakarta earthquake	182	18,607	19,202	23m 8s
2	Jakarta flood	346	28,929	34,091	25m 55s
3	Lombok earthquake	480	20,328	22,708	164m 36s

[7]https://github.com/yohanesgultom/respondor

V. Conclusion

The proposed system was able to generate route subnetwork for all operations within considerable processing time. In the future, we suggest to implement the process of finding minimally required routes and nodes to run on Graphics Processing Unit (GPU) so that the processing time can be more scalable.

References

[1] M. Haklay and P. Weber, "Openstreetmap: User-generated street maps," *IEEE Pervasive computing*, vol. 7, no. 4, pp. 12–18, 2008.

[2] A. Gemsa. (2017) Osmtoroadgraph. [Online]. Available: https://github.com/AndGem/OsmToRoadGraph