Access to Prosperity
Quantifying Infrastructure Impact with OSM
Quantifying Infrastructure Impact with OSM

- **Why**: Community development projects in eSwatini
- **How**: Using data analysis with OSM to help maximize impact
- **What**: Routing analysis with *tebetebe* python package
Where is eSwatini (Swaziland)?
Where is eSwatini (Swaziland)?
Where is eSwatini (Swaziland)?
PIGG’S PEAK – Ntfonjeni Member of Parliament Sompisi Magagula and some residents of Mganda cross the flooded Mlumati River. The situation has seen some pupils of Lufafa Primary School not being able to go to school for about two weeks. SEE PAGE 4 (Pic: Joseph Zulu)
https://youtu.be/RT_MPrKsCMM
Now What?
Now What?

- Develop Sustainable Partnership Structure
- Perform National Needs Assessment
Now What?

- Develop Sustainable Partnership Structure
- Perform National Needs Assessment
Where is there a need for footbridges in the country?
Footbridge Needs Assessment: Old Way

Pros:
- It works

Cons:
- Time Consuming
New Needs Assessment: Powered by OSM

1) Visit Tinkhundla Meetings
   • Present project to local leaders
   • Record lon/lats of potential bridges (QGIS)
Inkhundla Meetings: (QGIS)
- Potential Site lon/lat
- Destination POIs
- Flood Prone Bridges
New Needs Assessment: Powered by OSM

1) Visit Tinkhundla Meetings
   • Present project to local leaders
   • Record lon/lats of potential bridges (QGIS)

2) “Travelling Salesman” Site Visits
   • On-site assessment trip (OsmAnd)
   • Social & technical survey (ODK)
Site Visits: (ODK & OsmAnd)
- Social & Technical Survey
- Collect Site Media
Site Visits: (ODK & OsmAnd)

- Social & Technical Survey
- Collect Site Media
New Needs Assessment: Powered by OSM

1) Visit Tinkhundla Meetings
   - Present project to local leaders
   - Record lon/lats of potential bridges (QGIS)

2) “Travelling Salesman” Site Visits
   - On-site assessment trip (OsmAnd)
   - Social & technical survey (ODK)

3) Footbridge Impact Analysis
New Needs Assessment: Powered by OSM

1) Visit Tinkhundla Meetings
   • Present project to local leaders
   • Record lon/lats of potential bridges (QGIS)

2) “Travelling Salesman” Site Visits
   • On-site assessment trip (OsmAnd)
   • Social & technical survey (ODK)

3) Footbridge Impact Analysis
Footbridge Impact Analysis

We have:
- Litany of potential sites
- Heaps of social & technical data

We want:
- Prioritized list of sites, ordered by impact

But...
- Collected impact data is unreliable
- We want a deterministic approach
Footbridge Impact Analysis
Routing Scenario Comparison:

1) Normal vs. Flood
   What impact does a flood have?

2) Flood vs. Bridge
   What impact does a bridge have in flood?
Normal vs. Flood
Normal Scenario
Flood Scenario
Flood Scenario
Bridge Scenario
Bridge Affected Flood Routes
<table>
<thead>
<tr>
<th>origin_id</th>
<th>dest_id</th>
<th>normal_dist</th>
<th>normal_time</th>
<th>flood_dist</th>
<th>flood_time</th>
<th>bridge_dist</th>
<th>bridge_time</th>
</tr>
</thead>
<tbody>
<tr>
<td>405361898</td>
<td>718997017</td>
<td>5730</td>
<td>4125</td>
<td>7285</td>
<td>5245</td>
<td>5753</td>
<td>4142</td>
</tr>
<tr>
<td>404537021</td>
<td>718997017</td>
<td>8829</td>
<td>6357</td>
<td>9002</td>
<td>6481</td>
<td>9002</td>
<td>6481</td>
</tr>
<tr>
<td>405361899</td>
<td>718997017</td>
<td>5732</td>
<td>4127</td>
<td>7287</td>
<td>5246</td>
<td>5755</td>
<td>4143</td>
</tr>
<tr>
<td>405361892</td>
<td>718997017</td>
<td>5527</td>
<td>3979</td>
<td>7082</td>
<td>5098</td>
<td>5550</td>
<td>3995</td>
</tr>
<tr>
<td>405374012</td>
<td>718997017</td>
<td>7766</td>
<td>5592</td>
<td>9321</td>
<td>6711</td>
<td>7789</td>
<td>5608</td>
</tr>
<tr>
<td>405361893</td>
<td>718997017</td>
<td>5565</td>
<td>4007</td>
<td>7120</td>
<td>5126</td>
<td>5588</td>
<td>4023</td>
</tr>
</tbody>
</table>

How much time / distance is saved for each homestead, on average?
tebetebe: routing analysis with OSM

tebetebe is a Python API to compile, serve, and query routable networks using the Open Source Routing Machine (OSRM) and OpenStreetMap data, and provides a framework for routing analysis using these networks.

Package Overview

tebetebe makes it easy to compile a custom routing Scenario by abstracting OSRM executables into a pythonic API and provides a framework for routing analysis. With the range of customization available in the .lua configuration scripts, specific, accurate and readable transportation models can be developed and analyzed.

tebetebe also simplifies the routing analysis pipeline by enabling data to be pulled live from the OSM via the Overpass API and providing various user-contributed classes which automate common routing analysis tasks, such as isochrones.

Installation

1. Install osrm-backend binaries

   - See the osrm-backend wiki for instructions on how to build and install from source.
   - A Python setup script is included which allows for easy installation of both the Python bindings and the OSRM executables.
Scenario Schematic

RoutingProfile
(ex. car, bicycle, foot, flood...)

OSMDataset
(ex. GeoFabrik extract)

Scenario

ScenarioAPI

with Scenario():

POIDataset

POIDataset
Scenario Schematic

- RoutingProfile
  - (custom): (ex. car, bicycle, foot, flood...)
  - (standard): (ex. GeoFabrik extract)

- OSMDataset
  - .osm file
  - Overpass Query

- Scenario
  - with Scenario():

- ScenarioAPI

- POIDataset
  - OGR Dataset
  - Overpass Query

- POIDataset
  - OGR Dataset
  - Overpass Query
Normal Scenario

FootProfile
(ex. car, bicycle, foot, flood...)

Swazi .pbf

SwaziFoot

with Scenario():

SwaziFootAPI

origins

Homesteads

dests

Schools
Flood Scenario

FloodProfile
(ex. car, bicycle, foot, flood...)

Swazi .pbf

SwaziFlood

with Scenario():

SwaziFloodAPI

origins

dests

Homesteads

Schools
Flood Scenario

FloodProfile
(ex. car, bicycle, foot, flood...)

Swazi .pbf

SwaziFlood

SwaziFloodAPI

with Scenario():

origins

dests

Homesteads

Schools
block bridges with `flood_prone=yes`

```python
function WayHandlers.no_flood_prone(profile, way, result, data)
    bridge = way:get_value_by_key("bridge")
    flood_prone = way:get_value_by_key("flood_prone")

    if bridge == "yes" and flood_prone == "yes" then
        return false
    end
end
```

block nodes with `ford=yes`

```python
local ford = node:get_value_by_key("ford")
if "yes" == ford then
    result.barrier = true
end
```
from tebetebe.profiles import foot
import tebetebe as tb

tb_env = tb.Environment(tmp_dir="./tmp/simple_scenario")

mbabane = (31.1367, -26.3054)
simunye = (31.9274, -26.2108)

## Initialize scenario using eSwatini GeoFabrik extract and default foot profile
scenario = tb_env.Scenario("./tmp/swaziland-latest.osm.pbf", foot)

## Compile and run scenario
with scenario() as api:
    ## Query OSRM HTTP `simple_route` service to calculate route
    route = api.simple_route(simunye, mbabane)

    duration = route['routes'][0]['duration'] / 60
    distance = route['routes'][0]['distance'] / 1000

    print("Walking from Simunye to Mbabane")
    print(" Duration: {:.2f} minutes".format(duration))
    print(" Distance: {:.2f} km".format(distance))
from tebetebe.profiles import foot
import tebetebe as tb

tb_env = tb.Environment(tmp_dir="./tmp/simple_scenario")

mbabane = (31.1367, -26.3054)
simunye = (31.9274, -26.2108)

## Initialize scenario using eSwatini GeoFabrik extract and default foot profile
scenario = tb_env.Scenario("./tmp/swaziland-latest.osm.pbf", foot)

## Compile and run scenario
with scenario() as api:
    ## Query OSRM HTTP `simple_route` service to calculate route
    route = api.simple_route(simunye, mbabane)
    duration = route['routes'][0]['duration'] / 60
    distance = route['routes'][0]['distance'] / 1000

    print(f"Walking from Simunye to Mbabane")
    print(f"Duration: {format(duration)} minutes")
    print(f"Distance: {format(distance)} km")
tebetebe Features

- OSRM Scenario API
- Analysis Plugins
  - Accessibility Isochrones
  - Route Comparison
  - Parallel Scenarios
- Routing Analysis Pipeline
  - Analysis Environment
  - Overpass Integration
tebetebe Wish List

- Implement Shared Memory
  - Route network manipulation without recompilation
- Incorporate Osmium filters
  - Allow programmatic edits of route networks
  - ex. upgrade all tertiary roads to paved
- Parallelization
  - Asynchronous to multiple scenarios
- Remove python-osrm dependence
  - Streamline communication to HTTP API
- QGIS Plugin
  - Bring routing analysis to the masses
tebetebe Caveats

- **HTTP API means a lot of overhead**
  - If performing bajillions of single requests, this can be a bottleneck

- **File System I/O**
  - OSRM reads OSM data from disk and writes scenarios to disk
  - Not ideal in some circumstances (planet.osm)

- **Default routing profiles may not be accurate for your use case!**
  - Accurate routing requires customization
Check out the Docs!

https://1papaya.github.io/tebetebe/

- Script Examples
- API Documentation
Acknowledge Mints
Acknowledge Mints

You!!!
200 Kids Map Swaziland for Malaria Elimination

Posted by Cristiano Giovando

Working Groups

Associated Projects
Missing Maps

On Thursday, March 6th 2014, a team of 200 kids gathered in a library near the capital city of Mbabane to help create a map of the malaria prone areas of the country. The children were divided into teams and each team was assigned a sub area within Swaziland to gather data. The kids were taught how to use Google Earth to create a map of their area and then use that map to mark the areas where they had found evidence of malaria. The kids then used their maps to identify areas where more malaria prevention efforts are needed.
Modern C++ routing engine for shortest paths in road networks.
Thank You!!!

Dankeschön!
Muchisimas Gracias!
Merci!
Siyabonga Kakhulu!
Ke a leboha Haholo!
благодарю вас!
Salamat!
Sko Buffs!