

U2T1

Avaliando
algoritmos para o
caminho mais
curto em grafos
urbanos



Objetivo

Avaliar e comparar três algoritmos de menor caminho (OSMnx, Dijkstra tradicional e Dijkstra com min-heap) aplicados a um dos cenários urbanos propostos, com foco em desempenho computacional, similaridade das rotas e impacto ambiental (pegada de carbono).

similaridade das rotas*

Verificar se as rotas encontradas são semelhantes

Bibliotecas/Ferramentas/Algoritmos

Week 06

- **PDF** **Small World Cont.:** Paths, Distances, Connected Components, Clustering Coefficient, Social Distance and Six Degrees of Separation
 - **Notebook** Paths, Walks and Distances
 - **Notebook** Connected Components
 - **Notebook** Clustering Coefficient
- **PDF** **Classical Algorithms: Dijkstra:** Shortest path algorithm
 - You will learn: a) Explain how the Dijkstra algorithm works. b) Understand the algorithm's time complexity.
 - Ready to practice?
 - **Notebook**: **Dijkstra**: Implement the algorithm both with and without path reconstruction.
 - **Notebook**: **Min-Heap**: an implementation of Dijkstra's algorithm using a min-heap, with and without path reconstruction.



gboeing / osmnx

Type / to search



<> Code Issues 3 Pull requests 1 Actions Security Insights



osmnx Public

Watch 109

Fork 834

Starred 5.1k



main



3 Branches



75 Tags



Go to file



<> Code



gboeing

Merge pull request #1290 from gboeing/fix



05381b1 · 3 weeks ago



4,222 Commits



.github

pass args with equal sign

2 months ago



docs

update docs

3 months ago



environments

parallelize pytest with pytest-xdist

2 months ago



osmnx

note sorted by importance in docstring

3 weeks ago



tests

update query to test a TypeError

3 weeks ago



.gitignore

update gitignore

4 months ago



.pre-commit-config.yaml

update ruff

3 weeks ago



CHANGELOG.md

Update CHANGELOG.md

3 weeks ago



CITATION.cff

update citation

last year

About

Download, model, analyze, and visualize street networks and other geospatial features from OpenStreetMap.

osmnx.readthedocs.io

python

mapping

openstreetmap

osm

transportation

geospatial

routing

gis

spatial

urban-planning

transport

networkx

networks

spatial-analysis

overpass-api

spatial-data

geography

street-networks

urban

osmnx

Readme

MIT license

Cite this repository

codecarbon

Public

Sponsor

Watch 22

Fork 204

Starred 1.4k

master

147 Branches

53 Tags

Go to file

Add file

<> Code

 prmts128 fix(carbonserver) validate fief token (#825) ✓

018cc95 · 20 hours ago

2,138 Commits

.conda	Add 3.13 in variants.yaml	7 months ago
.github	Remove duplicate PUE (#823)	3 weeks ago
act	API for codecarbon (#145)	4 years ago
carbonserver	fix(carbonserver) validate fief token (#825)	20 hours ago
codecarbon	bump	4 days ago
dashboard	feat(core): ✨ allows picking up API endpoint from conf fil...	last year
deploy	Fix RAPL path	last week
docker	Remove tox and makefile mentions	last year
docs	bump	4 days ago
examples	remove percpu	3 weeks ago
requirements	bump deps	3 weeks ago

About

Track emissions from Compute and recommend ways to reduce their impact on the environment.

mlco2.github.io/codecarbon

Readme

MIT license

Activity

Custom properties

1.4k stars

22 watching

204 forks

Report repository

Releases 48

v3.0.1

Latest

4 days ago

+ 47 releases

Cenários de Avaliação

Roteamento para serviços de emergência

Contexto: Avaliar o tempo de resposta de ambulâncias saindo do Hospital Walfredo Gurgel para diferentes bairros.

- Origem fixa (hospital)
- Vários destinos (bairros populosos como Felipe Camarão, Alecrim, etc.)
- Comparar: tempo e distância com Dijkstra ($O(n^2)$), Dijkstra com heap, e OSMNx

Transporte público e deslocamentos urbanos

Contexto: Avaliar as rotas entre terminais de ônibus (ex: Terminal do Soledade → UFRN).

- Você pode comparar os algoritmos para identificar gargalos e ineficiências
- Relevância: deslocamentos de estudantes

Acesso à UFRN a partir de diferentes bairros

Contexto: Medir a acessibilidade da UFRN para estudantes vindos de bairros como Cidade da Esperança, Potengi, ou Pajuçara.

- Poderia incluir até simulações com interdições de ruas (ex: obras ou alagamentos)

Rotas de entrega por motoboys

Contexto: Simular o trajeto de entregadores que saem da Av. Prudente de Moraes para regiões como Neópolis, Candelária, ou Capim Macio.

- Tempo é fundamental → ideal para algoritmos que minimizam o custo/tempo com pesos reais

- Critérios de Avaliação
 - Tempo de execução
 - Similaridade e sobreposição de rotas
 - Pegada de carbono (usar a biblioteca codecarbon)
- Visualização e análise
 - Gerar mapas com as rotas
 - Construir tabelas comparativas com as métricas de avaliação
 - Discutir eventuais diferenças nos caminhos gerados

Entrega

Organizar o projeto em um repositório no GitHub

- README.md explicando o cenário escolhido, metodologia, resultados e conclusões
- Código limpo e comentado
- Figuras e tabelas bem formatadas

Vídeo explicativo de até 10 minutos (pode ser hospedado no Loom ou YouTube, link no README)



Data de entrega: 12 de maio (segunda-feira as 23h59)



Trabalho em dupla ou individual

3,00 (três pontos) na Unidade 2.