Carbon Link – Weighted Supply Chain Route Planning

By Jacob Igo

07/22/25

This tool is designed to give supply-chain managers a leg up by giving them real-time routing between their distribution locations. It considers the type of transportation used and calculates the carbon footprint generated from place to place using a factor of their distance. This allows us to not only calculate the fastest route based on distance, but the most efficient route based on carbon output from site to site. This tool also comes equipped with a personalized local AI model that takes in the user’s routing data and gives suggestions on how to uniquely plan and organize trips.

Setup

These are the required packages that need to be installed:

Backend:

* Python >= 3.11
  + Flask, flask\_cors, networkx, neo4j\_utils, requests
  + To install: pip install <package name>
* Neo4j
  + Make a free account on the website, then register a free AuraDB instance. Take note of the uri, your id, and token. For reference: <https://neo4j.com/docs/api/python-driver/current/api.html>
  + This is where you will manage your locations and connections between them
* Ollama
  + Download Ollama from their website: <https://ollama.com/>
  + Open up a terminal and run “ollama pull llama3.2”
  + This will give you a local AI model running on your computer

Frontend:

* Node.js
  + React, Vite, mapbox-gl, Cytoscape
  + To install: npm install / npm install <package name>

To start this project, open the zipped file of the code and open it. Before running it, PLEASE BE SURE TO CHANGE THE Neo4jHandler CREDENTIALS TO YOUR OWN FOR YOUR OWN DATABASE.

Next, open up three terminals. For the first one, navigate to the backend directory, and run python app.py. For the second one, navigate to the frontend directory, and run npm install, then npm run dev. For the third one, run the command ollama run llama3.2.

Now, in your browser, go to the URL that the frontend terminal says your project is running on. Your project should be live.