Data Science Technical report - Greener HSL

Ali Onur Özkan, Erik Olsson, Aki Kankaanpää

October 2024

1 Introduction

Our group attempted to create an application that would help public transport users track their carbon dioxide emissions based on the Helsinki Regional Transport Authority (fin. 'Helsingin seudun liikenne', HSL) routes that they take. This would provide environmentally conscious consumers the opportunity to be able to see, concretely, how their choices can help conserve our delicate climate. Databases, including the carbon emissions and bus numbers and routes, were used as the basis for our algorithm. Our teams end product is an app, which takes the start location and end location as arguments, and provides the user with three alternative routes, listing the emissions and travel duration. The final name for this app was 'Greener HSL (gHSL). This app was delivered to users via a website, which was launched on the 11th of October. Moreover, our team performed data analysis on the emission data, to understand trends on the various types of vehicles currently used by HSL.

2 Background

From the first meeting, our team was set on using data provided by The Helsinki Regional Transport Authority (HSL) as our raw data. Two datasets which piqued our interest were bus emission data, as well as route data for each bus type. All of this data was provided open-source by HSL, and was handled with respects to local data security and fair use laws.

After a meeting of back and forths, our team settled on the idea of both visualising the carbon emissions of different modes of transport, as well as developing an app where the information could be used to educate others.

By the end we achieved both goals and feel satisfied in our mini-project. gHSL was what we had hoped and the visualisations offer meaningful information not found elsewhere.

3 Data Cleanup

The data provided by HSL is stored in long '.txt'-files, where various sets of information (some of it missing, NULL) are separated by commas and line breaks. For this reason, our team chose to use python for data analysis, as the data was the easiest to process in the language.

Utilising Visual Studio Code, and the VS Code LiveShare plugin to work collaboratively in real time, the first step we took was combining the data from two separate files (emissions.txt, routes.txt) into a combination '.csv'-file. This made the rest of data cleaning far easies.

Once all of the data had been stored in the same place, duplicate rows such as 'routeid' (literally just a numerical representation of the route) were deleted. Additionally, each NULL datapoint was checked, and each was deemed to be an out-of-service or seasonal route, thus they could be removed.

By the end of data cleanup, we had taken two '.txt'-files both containing 453 lines into a clean dataset of 424 columns and 10 rows for a total of 4240 datapoints. We chose to keep certain data as not to sanitise the information too much, in case a user was interested in finding out more about our data. This '.csv'-file is packaged together with our route-emissions app.

4 Program Development

Initially the plan was to calculate route distances and CO2 emissions, perform the carbon emission calculations and find the greener route by ourselves. However, since there was no data provided on the distances the busses traveled to get to the next stop it proved impossible. Changes had to be made to the plan so that route information could be incorporated into the app. We decided to use a routing API provided by HSL, paired with our preprocessed data, which allowed for route distance emission calculations.

4.1 Initial Canvas

```
Title: HSL Route Emissions
MOTIVATION: Targeted towards Eco-Conscious Public Transport Users, and we must focus
on emissions. The solution can help users pick more eco-friendly routes.
DATA COLLECTION: Data to be used is from HSL (Open Data, GTFS-RT APIs), and it is
stored in .txt-files. We plan to implement the data in a program to help transport
users. We are stitching together at least two separate databases.
PREPROCESSING: Preprocessing will primarily focus on combining data from different
files into an easy-to-read database, which can then be implemented in the ways in
which we want to (Emissions and routes are stored as different databases)
→ Maybe use GAdmin4 to view the data.
EXPLORATORY DATA ANALYSIS (EDA): After assessing the data, we plan to find possible
connections to things outside of the data, but more importantly, to understand what
the data directly implies about HSL emissions (connections, patterns).
VISUALIZATIONS: Hopefully a simple UI with a search function and possibly an
emissions calculator.
LEARNING TASK (focus on problem definition): Problem setting is HSL route and
emissions data, and they also make up the target variables. Our goal is to
understand route data analysis.
LEARNING APPROACH (focus on solution implementation): Most relevant methods
for this project are most likely going to be organizing data, and assess means
and ratios (simple calculations, with tremendous implications).
COMMUNICATION OF RESULTS: An app with a bare-bones UI will be most sufficient
(alternatively a database could work as well).
DATA PRIVACY AND ETHICAL CONSIDERATIONS (if applicable): N/A
ADDED VALUE: We can consider ease of access as added value, as the data we
will be using is open access, and thus any consumer could browse through it.
If we have time, some sort of 'route emissions' calculator could be implemented.
```

4.2 Feedback and Changes

Our helpful project instructor provided our team the following feedback: *The project seems very simple. Approach is unclear to me. Hopefully you provide a thorough analysis of the findings and a webapp that gives the most ECO-friendly routes. The time of the route should also be included, as the user probably wants to know both and if there is a tradeoff between the two.*

Based on this feedback, we tried to improve our project by both opening up our approach in our technical report, as well as provide clear justification for the existence of our app when presenting the project. It was clear that we needed to condense our plan into concrete ideas to provide to users.

What should be presented to potential users:

- Compare transportation efficiency for different types.
- Effect of increasing passenger count on carbon emissions.
- Sustainability recommendations (which routes should be favored)

With these improvements in mind, we moved on to analyse data and to create the app.

4.3 Data Analysis

With a clean table, it was intuitive to use the pychart plugin to create clean looking charts. The data we had contained routes, transportation type, CO2 per km, passenger count. The following two charts were created based on that information, our assessment of what we intrigued us personally. Both were included in the presentation showcasing our project to potential users:

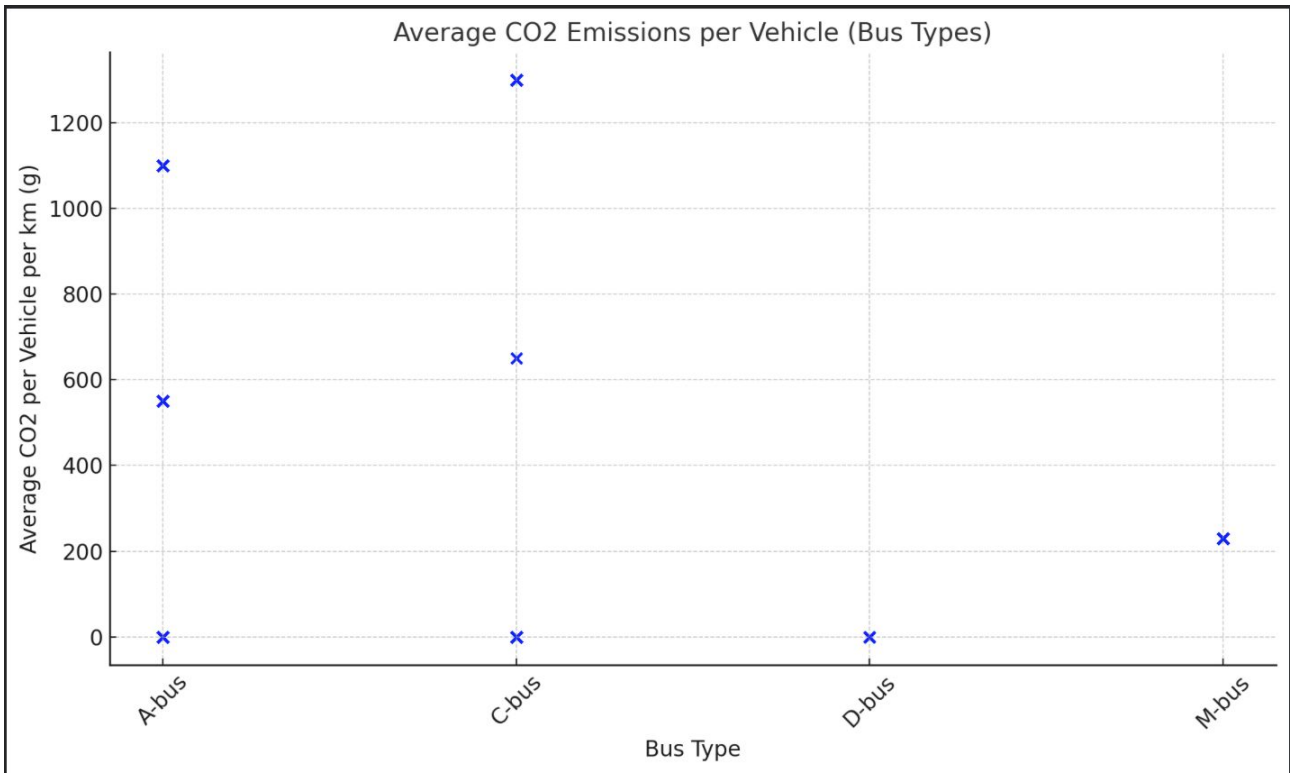

Figure 1: The first plot-chart visualises the fact that not only does HSL use different types of buses, but that same types of buses have variation in emissions between separate models

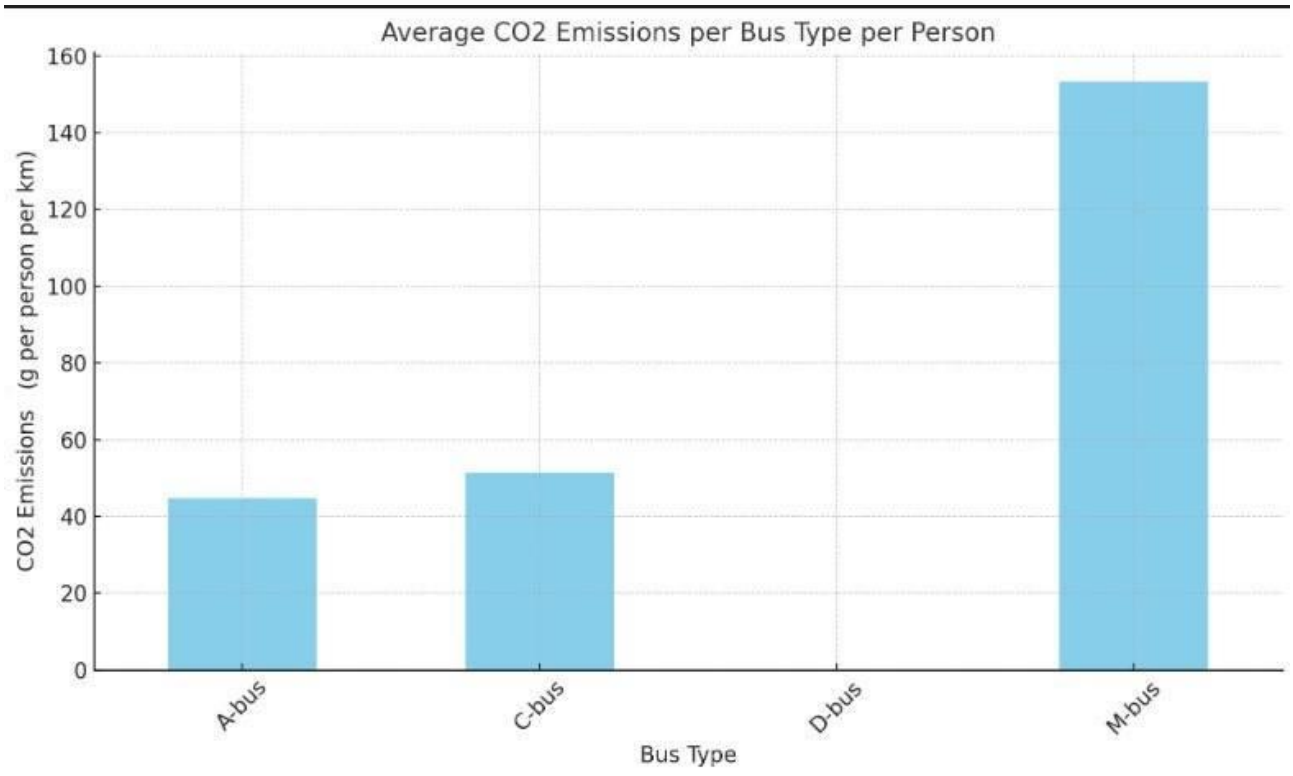

Figure 2: The second bar chart emphasises that it is important to consider the load capacity of different bus types, when discussing enviromental impact of different forms of transport.

4.4 The Emissions App

We had the routes paired with their emissions. In addition to this we also had a coordinates map of different stops, provided by the HSL API.

While two of our group members worked on the back-end implementation of the App, one of us worked on both the front-end of the app, and programming the website in HTML and CSS.

The back-end developers essentially created a program, which stores two inputs from the user (start and end), passes them through the HSL API (via the returns addon for python), locating the potential routes, then each mode of transport is stored in a python dictionary, along with how many km the vehicle is taken for. For each item in the dictionary, the corresponding mode of transport (routename) is fetched from our 'csv.'-file, along with its emission data. This data is returned to the user, providing a calculated emission based on the distance times emissions per kilometer.

On the front-end side, the python addon Tkinter was used for the GUI. Thus the program could be converted to an executable file, to ease the use of the app. The user fills out two boxes, presses one button, and receives a clearly formatted set of results. Based on demo feedback, users felt that the app was clear an streamlined, and thus nothing to break visual clarity was added.

A simple website with a button to download the necessary files was included, along a simple breakdown of instructions on how to get started. The site can be accessed via: https://tinyurl.com/3h697dkc

5 Reflection

5.1 Challenges

The biggest issue we felt we could not truly address was the scope of our project. All steps of the data lifecycle are present, however we were lacking on the interpretation of the data in our first canvas. This issue was addressed by presenting users with meaningful data.

A second challenge was bringing the HSL API and our data together. It took many brainstorming sessions to find a method which worked. We were very proud of finding a way do this!

5.2 Successes

Our team felt success at our ability to combine our preprocessed data with an API provided by HSL. This makes the calculations made in the program quicker, and less hardware demanding for the user.

Overall we are proud of how well the three of us were able to come together and create something functional to present to other students.

As we are all second year bachelor students, this has been one of the largest collaborative programming projects in our degree thus far, and it definitely taught us a ton of meaningful information about a reasonable workflow and team effort.

Time management was also never an issue, and our group chose a topic which interested us all, helped us lean into our strengths and challenge ourselves.

5.3 Key Pointers

While the topic was quite original, considering that we couldn't find any app about the eco friendliness of public transport routes in Finland, our group did note that many other groups were also focusing on either HSL raw data, or emission calculations. This was not a huge issue for us, as we felt we provided something meaningful to users.

Each group member additionally felt that the course homework helped with the project, especially pychart and pandas. In fact we are very happy with how the course was structured, and each week seemed to answer any issues or worries we might have had about how to continue with your project.

5.4 Presentation and The Future

In our presentation we wanted to emphasise that while public transport produces carbon emissions, it is eco-friendlier thanks to load capacities. Additionally we noted that certain vehicles, such as the metro-lines and trams do not produce carbon emissions, as they run on electricity.

We wanted to remind individuals that in our daily lives, we can be more aware of the emissions we produce. While we may state: 'The bus will drive the route no matter if I get on it or not' but as we have seen before, by choosing the eco-friendlier vehicles or routes, HSL will lower more carbon-heavy routes, and work on going greener.

The usefulness of the project was clear to the user, the fact that you are now able to check the emissions that a route produces gives users of public transportation a way to individually control their own emissions.

Our user demographic is limited: anyone outside of the Helsinki transportation area would not have any use for it. If you use a car, you would not have much use for it either. This is something that could be expanded upon in the future, perhaps implementing world-wide transport networks to help users across the globe battle climate change.

Additionally, while one of our group members used a macbook, gHSL currently only works natively on Windows and Linux, but future Mac support is on its way!

6 Conclusion

Our team created an app, which a user of public transport can open to check how significant is their carbon emission for a given route. Through collaboration, problem-solving, and using the course material as our guide, our group gathered raw data from HSL, combined and processed two large datasets into a singular, clear to read format.

The final product was made to be an extremely easy to navigate and use app. There are only two text boxes, current position and destination, and a button to confirm choices, the results are easy to understand but simultaneously give a detailed and well structured response.

The app was presented to potential users among with analysis of the data to provide a comprehensive and persuasive speech.

The entire purpose of this project was to create something that not only works, but that people could feel inclined to use. Since climate change is a timely topic, anything that directly aims to help climate conscious people will be useful in real world applications.

The inability to check carbon emissions on public transportation routes provided a challenge we sought to solve, and the implemented approach is very efficient. Since the project incorporates the use of the HSL route API and off-site processing, it makes the ease of use, security, and reliability as good as any service HSL provides.

Overall our group is satisfied with the final result and what steps we took to get there.