

# Assignment 5

## Evaluating Environmental Impact of Exam Portfolio

*Date: 02/05/2024*

Laura Givskov Rahbek

### Description

This folder contains assignment 5 for Language Analytics. The objective of the assignment is to think critically about the effects and impact machine learning has on the environment, to write code that can extract approximate benchmarks showing this impact and to present and discuss the results in an understandable way. More specifically, the package **CodeCarbon**'s class **EmissionsTracker** is utilised to measure the approximate  $CO_2$ -equivalents ( $CO_{2eq}$ ) in kg for assignments 1-4 in this repository. The resulting values are used to investigate the environmental impact of the different tasks performed in the four previous assignments done in this class.

The **assignment-5** folder contains:

- An **out** folder which contains the .csv files with the emission data from each of the four assignments.
- A notebook **emissions.ipynb**, where the data wrangling and visualisation was done.

### Usage and Reproducing of Analysis

All code used to handle the emissions data as well as visualisations of these can be found in the notebook. If relevant the packages needed to run the **emissions.ipynb** notebook can be installed by running **bash setup.sh** in the terminal. However, the dataframes and plots are visible in the notebook as is, without needing to run it.

To track the emissions from the assignments the **EmissionTracker** from **CoddeCarbon** was used, to see how precisely this was done, the code can be found in the scripts in the assignments. Essentially, the tracker was initialised and then used to track different tasks, to be able to investigate the individual tasks as well as the overall impact of each assignment. The emissions were added up to infer the total emissions per assignment.

## Discussion

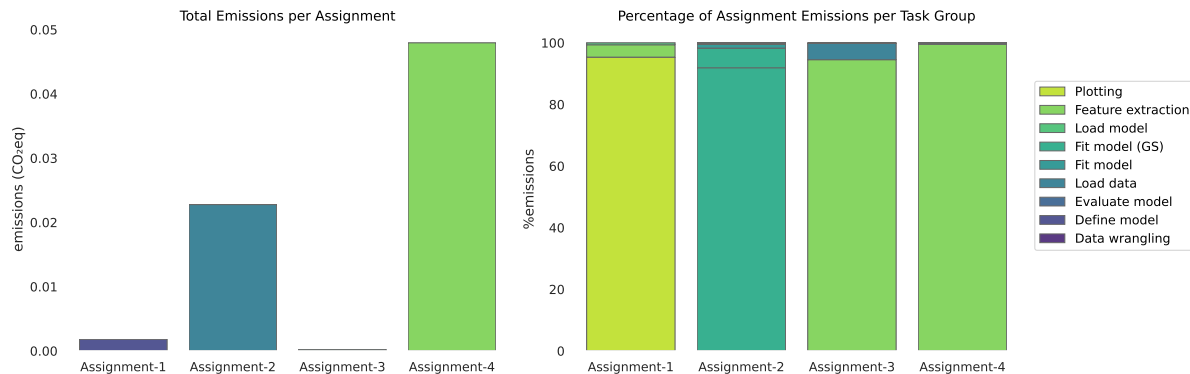
Before discussing the results of the environmental impact analysis, it is worth noting that these values are estimates and approximations to the actual emissions caused by the code run in these assignments. Nonetheless these approximations are a useful representation, and are used to base the following discussion of the cost of machine learning off of. Firstly, none of the code, analysis or machine learning conducted for this exam is of great importance, except for learning, which of course is why it has been run and made. I would not be able to argue that it is worth the substantial amount of  $CO_2eq$  to analyse the emotional differences during Game of Thrones seasons, based only on the scripts. Considerations discussed are therefore concerned with the general use of machine learning, and the considerations when e.g. implementing grid search or using a much more expensive model that only does marginally better than a less expensive one.

### *Total Emissions for each Assignment*

Assignment	Emissions ( $CO_2eq$ )
Assignment 1	0.001807
Assignment 2	0.022768
Assignment 3	0.000123
Assignment 4	0.047947

The above table showcases the total emissions (in  $CO_2eq$ ) for each of the four assignments. Assignment 4 has the largest impact, followed by Assignment 2, Assignment 1 and lastly Assignment 3. Starting with the least impactful one: Assignment 3. In Assignment 3 task with the largest impact is the loading and usage of the word-embedding model `glove-wiki-gigaword-50`. It is used to identify ten of the closest (most similar) words to a given keyword - but that is it. It is pretrained, and it used for a very specific task, resulting in a comparatively low total emission for the assignment. Assignment 1 is the second least impactful, with the task of feature extraction, which consisted of the use of a `spaCy` model extracting and identifying six different types of tokens (parts of speech and named entity recognition). However, Assignments 1 and 3's impact is difficult to compare to the impact of both Assignment 2 and 4, as seen in the two plots below. The most impactful assignment is Assignment 4; where the primary task causing this level of emissions is the use of a pretrained large language model, handling a large amount of text. The second most impactful assignment is Assignment 2; where a large amount of the emissions comes from the gridsearch performed on the Convolutional Neural Network Classifier from `scikit-learn`. As mentioned in Assignment 2, the performance of the model resulting from the gridsearch was not improved in a way that justifies this amount of emissions.

*To the Left: Total Emissions per Assignment, To the Right: Task with Highest Emission per Assignment*



*The plot to the left shows the total emissions from each of the assignments and the plot to the right show emissions from the task emitting the most in each assignment. As discussed it is the use and training of large models that primarily drives the emissions*

As touched upon above, for each of the assignments it is one task that is responsible for most of the emissions for that assignment. The horizontal lines, visible on some of the bars representing emissions for each assignment in the plot above, shows emissions for the most impactful task for that assignment. It is the use and training of large machine learning models that overwhelmingly costs the most. However, as seen on the plot below, the difference in emission rate (CO<sub>2</sub>eq/second) is minimal between all the tasks - based on this

### plot with rate or duration included??

As discussed above, comes with finetuning large models, like the MLP neural network in Assignment 2.

- How robust do you think these results are and how/where might they be improved?

*change code to have less enviornmental impact*

Task discussion:

- Which specific tasks generated the most emissions in terms of CO<sub>2</sub>eq? Again, explain why this might be.