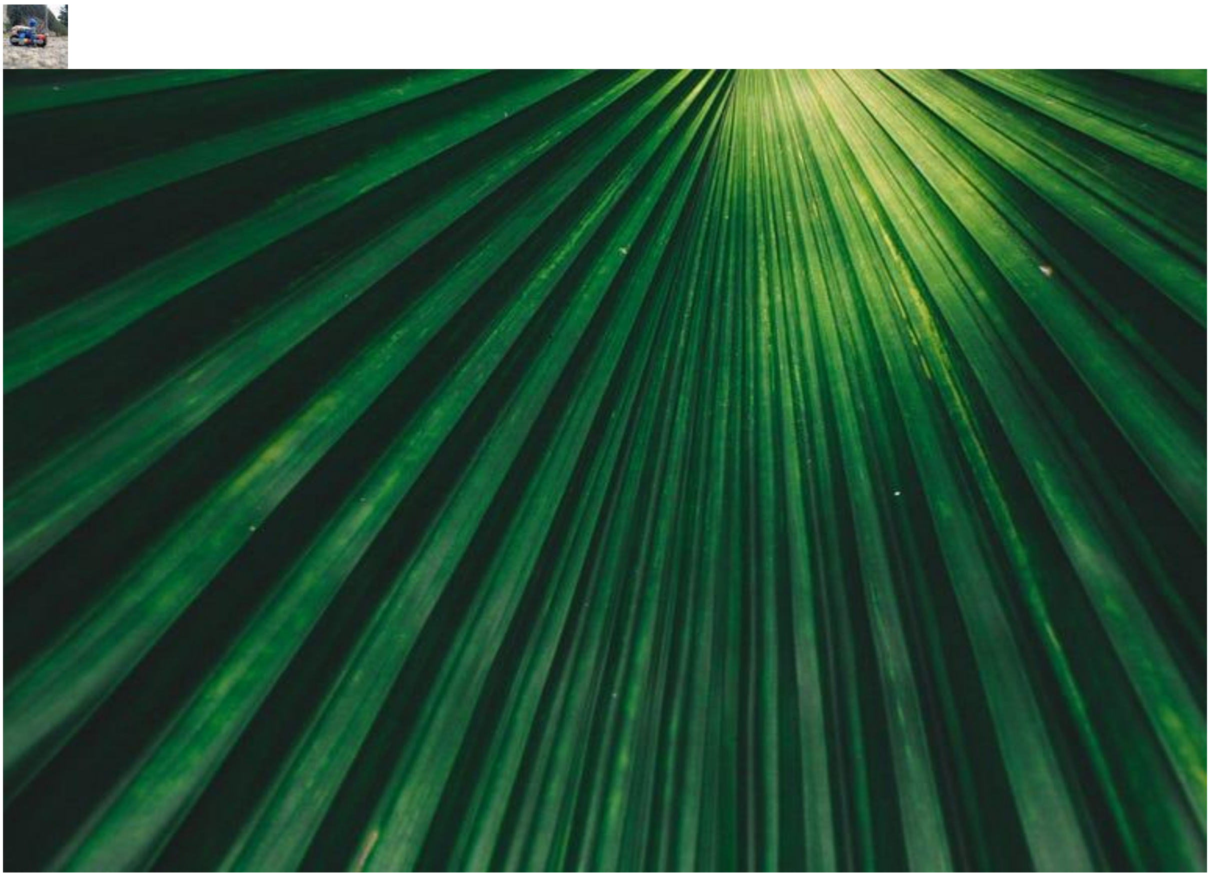


AI Computing Emits CO_2. We Started Measuring How Much.

Kamal Goyal

·
Follow
Published in

GAMMA—Part of BCG X

·
7 min read
·
Nov 30, 2020
108
3

It can be hard to appreciate just how much our modern world depends on vast amounts of computing power. Data and algorithms are increasingly embedded in nearly every aspect of modern life, from helping physicians detect cancer earlier to personalizing the advertisements consumers receive. And as the use of large-scale computing continues to increase, so does its impact on the planet. It is no small irony that the same technology helping scientists create greener energy is also consuming energy at a prodigious rate. According to some estimates, datacenters consume from 1% to 2% of all the energy generated each year around the world, with this amount increasing annually. Some tech companies boast that they are improving the efficiency of their datacenters. But with global compute instances rising by as much as 550% in the last ten years [1], the amount of energy it consumes — and the greenhouse gas emissions (GHG) it releases — will continue to grow.

As wildfires rage in the western U.S. and flooding and drought become more severe around the world, the climate damage caused by GHG emissions is painfully evident. In response, researchers and engineers from four organizations have taken the lead by developing a tool that measures the environmental impact of artificial intelligence computing. Comprised of experts from MILA, a world-leading AI research institute in Montreal; GAMMA, BCG's global data science & AI team, Haverford College in Pennsylvania; and Comet, a meta machine learning platform, the team has developed a tool that enables developers to track the carbon footprint of the energy consumed as they run their algorithms.

Quantifying Carbon Emissions

The surge in computational power has made applying AI to real-world applications possible by executing compute-intensive tasks in a reasonable amount of time. The number of floating-point operations per second (FLOPS), a measure of computing performance, has steadily increased by a factor of 150 since 2004, from 100 GigaFLOPS in 2004 to 15 TeraFLOPS in the latest GPU models.

One consequence of this increase in computing is the heavy environmental impact of training machine learning models. A recent research paper has shown that an inefficiently trained NLP model using Neural Architecture Search can emit more than 626,000 pounds of CO_2 equivalent, which is about five times the lifetime emissions of an average American car [2].

The carbon intensity of computing is directly related to the quantity and source of electricity it uses, measured in grams of CO_2-equivalent (CO_2eq) per kilowatt-hour of power consumed. CO_2eq is a standardized measure used to express the global-warming potential of various greenhouse gases.

There are 3 major factors that drive the environmental impact of AI, and data scientists can have a direct impact on the rate of CO_2 emissions by making informed choices on each of these factors:

1. **Grid Energy Mix**: Electricity from the grid that the hardware infrastructure is connected to may be generated by a combination of different energy sources (coal, petroleum, natural gas, low-carbon fuels). The combination used can result in significant variation in the average emissions in a single region, ranging from between 20g CO_2eq/kWh in Quebec, Canada to 736.6g CO_2eq/kWh in Iowa, USA [3]. The choice of cloud server region where you run your algorithms is the single most important choice to limit the environmental impact.

2. **Compute Time**: Training a powerful machine-learning algorithm can require running multiple compute machines for days, if not weeks. For example, the fine-tuning required to improve an algorithm by searching through different neural network parameters can be especially computationally intensive, since all possible combinations of parameters are usually tested via grid search. For recent state-of-the-art architectures like VGG, BERT and GPT-3, which have millions of parameters and are trained on multiple GPUs (graphic processing units) for several weeks, this can correspond to a difference of hundreds of kilograms of CO_2eq.

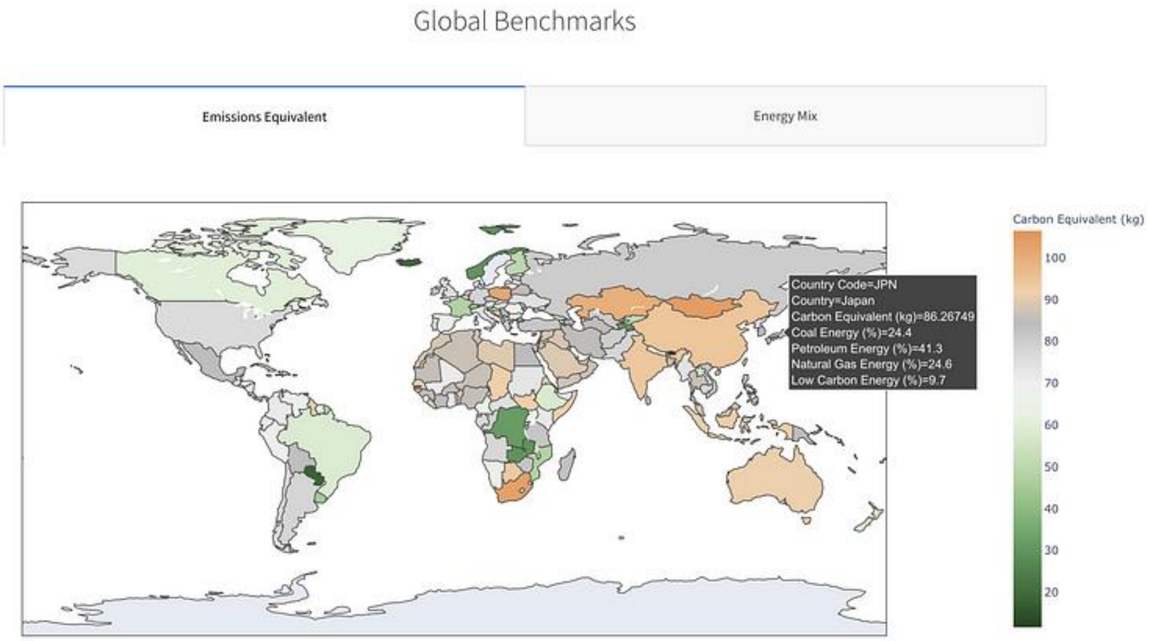
3. **Choice of Hardware**: Instead of using traditional chips, data scientists can reduce their environmental impact by turning to newer generations of computing hardware such as GPUs and tensor processing units (TPUs). These have been specifically designed for the parallel computations involved in training neural networks. Using this hardware can improve the efficiency of training ML models, reduce training time and energy and, therefore, reduce climate impact.

Using the Emissions Tracker

To track the carbon emissions of training AI/ML models, our team has built a tool called CodeCarbon. It comes as a light-weight pip package that seamlessly integrates into a Python codebase. As such, developers around the world who use Python can add this tool to their code and, with just a few more lines of code, start tracking CO2 emissions from the execution of the codebase.

```
from codecarbon import EmissionsTrackertracker =
EmissionsTracker()tracker.start()# GPU Intensive code goes
heretracker.stop()
```

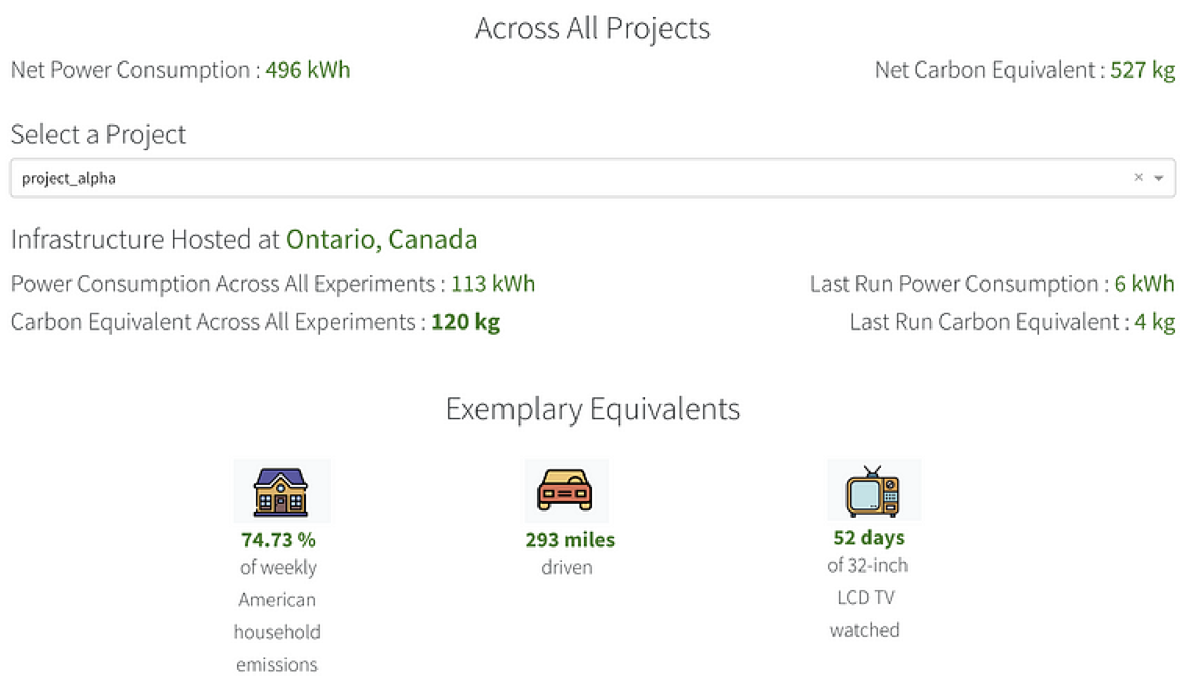
Here’s how it works: The tracker records the amount of power being used by the underlying infrastructure from both major cloud providers and privately hosted on-premise datacenters. Based on publicly available data sources, it then estimates the amount of CO2 emissions produced by referring to the carbon intensity from the energy mix of the electricity grid to which the hardware is connected.



Country Code=JPN
Country=Japan
Carbon Equivalent (kg)=86.26749
Coal Energy (%)=24.4
Petroleum Energy (%)=41.3
Natural Gas Energy (%)=24.6
Low Carbon Energy (%)=9.7

Equivalent Emissions and Grid Energy Source Mix

The tracker logs the estimated CO_2eq produced by each experiment then stores the emissions across projects that can be aggregated at an organizational level. This gives developers greater visibility into the amount of emissions generated from training their models and makes the amount of emissions tangible by showing equivalents in the number of automobile miles driven, the hours of TV watched, and the daily energy consumed by an average US household [5].



Across All Projects

Net Power Consumption : 496 kWh

Net Carbon Equivalent : 527 kg

Select a Project

project_alpha

Infrastructure Hosted at Ontario, Canada

Power Consumption Across All Experiments : 113 kWh

Last Run Power Consumption : 6 kWh

Carbon Equivalent Across All Experiments : **120 kg**

Last Run Carbon Equivalent : 4 kg

Exemplary Equivalents

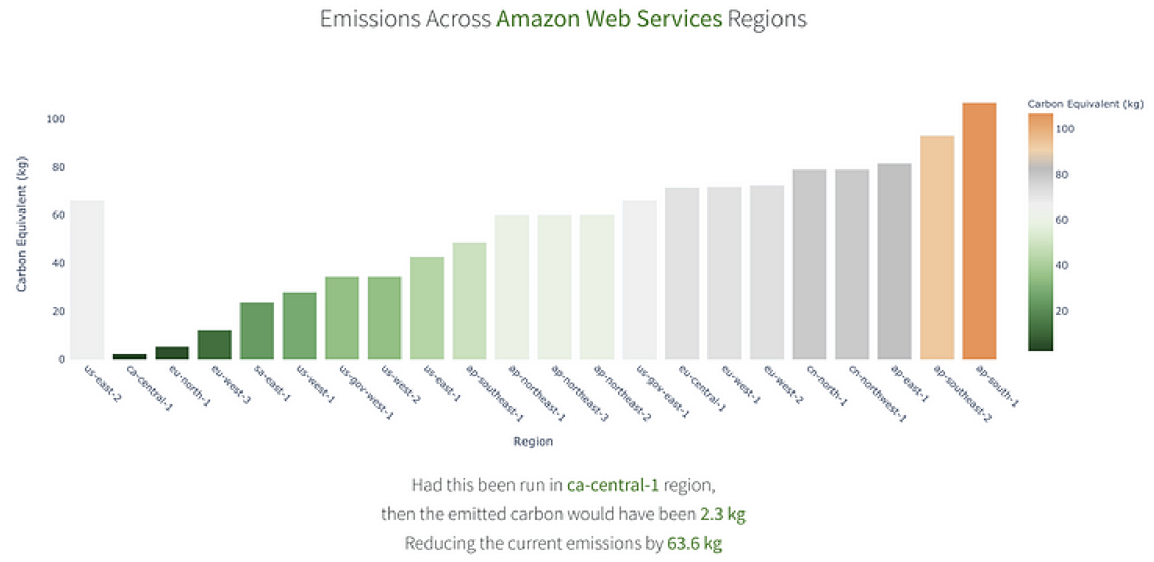
74.73 % of weekly American household emissions

293 miles driven

52 days of 32-inch LCD TV watched

Project Emissions and Exemplary Equivalents

The tool also has a dashboard that illustrates how the emissions footprint differs when the cloud infrastructure is hosted in different regions.



Emissions Across Amazon Web Services Regions

Had this been run in ca-central-1 region,
then the emitted carbon would have been 2.3 kg
Reducing the current emissions by 63.6 kg

Compare Emissions Across Different Regions

Scientia Potentia Est

Knowledge is power, so now that you know how to measure your carbon footprint, how can you reduce it?

Carefully choosing the computing infrastructure, cloud-server region, and efficient model-training practices will help reduce the carbon footprint of running your computational experiments. For instance, choosing a cloud provider region that uses low-carbon electricity is one immediate way to reduce your footprint.

Developers can also reduce the required training time by fine-tuning a pre-trained model rather than training a model from scratch. Recent research has shown that fine-tuning a pre-trained model for specific tasks in image recognition and NLP results in higher performance and requires lesser training data than fine-tuning a fresh model built from scratch.

While it is critical to arrive at an optimal hyperparameter configuration when training deep learning models, the widely used grid-search approach is extremely computationally intensive. Research has shown that random search is not only as efficient in terms of model performance but also accelerates the time to arrive at an optimal configuration, thereby reducing the net training time and hence the emissions produced [4].

Finally, the choice of computing hardware can influence the rate of computation delivered by a computer for every watt of power consumed, which is a measure of the energy efficiency of a computing machine. For applications in embedded systems where low-power consumption and efficiency are important, GPUs such as

Jetson AGX Xavier can be 10 to 20 times more efficient than traditional GPUs [3].

A Call for Action

The ability to track CO_2 emissions represents a significant step forward in data scientists' ability to use energy resources wisely and, therefore, reduce the impact of their work on an increasingly fragile climate. But our work to develop a way to measure the impact of artificial intelligence computing is just the first step. Our hope is that this tool will also help to introduce greater transparency into the developer community, enabling developers to measure and then report emissions created by a diversity of computing experiments.

To that end, we have designed CodeCarbon as an open-source tool. We look forward to developers and researchers using the tool and contributing to its improvement by enhancing it with new capabilities. We also encourage you to spread the word about the tracker among your colleagues and peers in conferences, on social media platforms, and at developer forums. To increase awareness of the environmental impact of computing, we highly recommend that you report the CO_2eq of your experiments in research papers, articles, and tech-blogs that you publish.

If recent history is any indicator, the use of computing in general and AI computing, in particular, continues to expand exponentially around the world. It is up to us to make sure our collective carbon footprint increases as little as possible.

References

1. Data-Center Power Consumption Holds Steady, Network World, March 10, 2020

2. Emma Strubell, Ananya Ganesh, Andrew McCallum. Energy and Policy Considerations for Deep Learning in NLP. arXiv:1906.02243, 2019

3. Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, Thomas Dandres. Quantifying the Carbon Emissions of Machine Learning. arXiv:1910.09700, 2019

4. James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. Journal of Machine Learning Research, 13(Feb):281–305, 2012

5. Kadan Lottick, Silvia Susai, Sorelle Friedler, and Jonathan Wilson. Energy Usage Reports: Environmental awareness as part of algorithmic accountability. NeurIPS Workshop on Tackling Climate Change with Machine Learning, 2019