How can we estimate the energy consumption of training an AI model?

Presentation and comparison of existing tools
Methodology
Energy consumption

Estimation
From hardware characteristics
(Thermal design power)

External measures
Power measures from outside the hardware

Software power models
From hardware performance counters
Taking into account external factors

\[ \text{Energy}_{\text{training}} = \text{Energy} \times \text{External Factors} \]

• With External Factors:
  • PUE: Power Usage Effectiveness (To take into account everything that is necessary for data centers to run, like cooling)
  • PSF: Pragmatic Scaling Factor (To take into account hyper parameter search)
From energy to carbon emissions

- **Energy** kWh
- **Carbon emissions** gCO2 eq
- **Carbon intensity** gCO2eq/kWh

\[ E_{\text{emissions}} = E_{\text{energy}} \times I_{\text{intensity}} \]
Use case for comparison

• **Application**
  - Handwritten digit recognition: Image classification
  - Model CNN: 2 convolutional layers, 2 fully connected layers (1 199 882 parameters, 5Mb)
  - Dataset: MNIST
  - Trained during 5 epochs (~ 120 seconds)
  - Library: Pytorch

• **Hardware**: gemini cluster (Grid'5000)
  - CPU: Intel Xeon E5-2698 v4 (Broadwell, 2.20GHz, 2 CPUs/node, 20 cores/CPU)
  - GPU: 8 x Nvidia Tesla V100-SXM2-32GB (32 GiB)

• **External factors**
  - Expected PUE: 1.58
  - Expected Carbon Intensity: 50 gCO2/kWh
Comparison of tools

• **Hardware estimations**
  - Green algorithms
  - ML CO2 Impact

• **External mesure**
  - Wattmeters

• **Based on software power models**
  - CodeCarbon
  - Experiment-impact-tracker
  - CarbonTracker
  - Energy Scope
Hardware estimation: Online tools
Green Algorithms (https://green-algorithms.org)

Energy = \text{runtime} \times ((\text{power}_{\text{CPUs}} + \text{power}_{\text{GPUs}}) \times \text{usage} + \text{power}_{\text{memory}})

\text{Energy}_{\text{training}} = \text{Energy} \times \text{PUE} \times \text{PSF}

\text{Emissions}_{\text{Carbon}} = \text{Energy}_{\text{training}} \times \text{Intensity}_{\text{Carbon}}

- No install
- Independent of the training
- Need to know the hardware
Hardware estimation: Online tools

**ML CO2 Impact**

[https://mlco2.github.io/impact/?#compute](https://mlco2.github.io/impact/?#compute)

\[
\text{Energy}_{\text{training}} = \text{runtime} \times \text{power}_{\text{hardware}}
\]

\[
\text{Emissions}_{\text{carbon}} = \text{Energy}_{\text{training}} \times \text{Intensity}_{\text{Carbon}}
\]

- No install
- Can be done before the training (or after)
- Limited (can only specify one hardware at the time and not memory)
- Global carbon intensity
- Can not specify PUE for private infrastructures
Green algorithm vs ML CO2 impact

Different maximal usage power?

Impact of carbon intensity source

Energy consumed given by the tools

Energy if PUE = 1.58

Figure 1: Energy: Given versus Expected

<table>
<thead>
<tr>
<th>Components</th>
<th>GA</th>
<th>MCI</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUE</td>
<td>1.67</td>
<td>1</td>
</tr>
<tr>
<td>Carbon intensity (gCO2/kWh)</td>
<td>39</td>
<td>432</td>
</tr>
<tr>
<td>Source</td>
<td>ElectricityMap 2020</td>
<td></td>
</tr>
<tr>
<td>Components</td>
<td>GPUs, CPUs, Memory</td>
<td>GPUs, CPUs</td>
</tr>
</tbody>
</table>

Table 1: Parameter values of GA and ML CO2 impact
External measurement

Wattmeters available on Grid5000

- **Wattmeters** on the power supply
  - One measurement every 20 milli-seconds
  - Precision at 0.1W

- Others (not studied here)
  - **PDU** (power distribution unit)
  - Intelligent Platform Management Interface (IPMI) & **BMC** controllers
    - One measurement every five seconds
    - Precision at 1W

- Exact value
- Independent of hardware

- Hard to install (except on Grid5000)
- At the plug level
- Expensive
Software power models

**RAPL interface**
- Power model from performance counters
- Model-Specific Registers (MSR) containing
  - Temperature, power, frequency, …
  - Per processes and components
- Only Intel Processor (Sandy Bridge and later)

**Nvidia SMI**
- Equivalent for Nvidia GPU
- Small granularity

**A lot of technologies are based on those models**
- Open-source python packages
  - CodeCarbon
  - Experiment-impact-tracker
  - CarbonTracker
- Profiling software
  - Energy Scope (Hervé Mathieu - Inria Bordeaux)
- Only on specific hardware
Python packages

- Quick install
- Good documentation
- Compute carbon emissions

- Fixed energy parameters
- Poor transparency
- Need privileged access

```python
import impactlib
# Loading dataset and processing it
tracker = impactlib.init()
tracker.start()
# Training
tracker.stop()
# Exporting results:
tracker.energy_consumption
tracker.carbon_emissions
```

\[
\text{Energy} = \sum \text{interval} \times (\text{power}_{\text{CPUs}} + \text{power}_{\text{GPUs}} + \text{power}_{\text{memory}})
\]

\[
\text{Energy}_{\text{training}} = \text{Energy} \times \text{PUE}
\]

\[
\text{Emissions}_{\text{carbon}} = \text{Energy}_{\text{training}} \times \text{Intensity}_{\text{carbon}}
\]
Profiling software
Energy Scope

- Good visualisation
- One data point every second
- Granularity per component
- Does not compute carbon emissions
- Still in development

\[ Energy_{training} = \sum interval \times (power_{CPUs} + power_{GPUs} + power_{memory}) \]
Real-time monitoring

Energy Scope granularity

Evolution of the power of both CPU during an experiment estimated by energy scope.

Evolution of the power of all GPUs during an experiment estimated by energy scope.
Comparison: Real-time monitoring

**Wattmeters vs Energy Scope**

Evolution of the power during an experiment from wattmeters and energy scope.
Comparison
Focus on energy

Figure 3: Energy: Given versus Expected
Impact of PUE
Unexpectedly low

Energy consumed given by the tools

<table>
<thead>
<tr>
<th>Components</th>
<th>W</th>
<th>CC</th>
<th>EIT</th>
<th>CT</th>
<th>ES</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUE</td>
<td>X</td>
<td>X</td>
<td>1.58</td>
<td>1.58</td>
<td>X</td>
</tr>
<tr>
<td>Components</td>
<td>Whole machine</td>
<td>GPUs, CPUs, Memory</td>
<td>GPUs, CPUs, Memory</td>
<td>GPUs, CPUs, Memory</td>
<td>GPUs, CPUs, Memory</td>
</tr>
</tbody>
</table>

Table 2: Energy parameter values

Figure 4: Tracking duration
Duration of each tool tracking
Comparison

Focus on carbon emissions

Impact of carbon intensity source

Figure 5: Carbon emissions: Given versus Expected

Table 3: Carbon emission parameter values

<table>
<thead>
<tr>
<th>Carbon intensity (gCO2/kWh)</th>
<th>W</th>
<th>CC</th>
<th>EIT</th>
<th>CT</th>
<th>ES</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>242.6</td>
<td>47.60</td>
<td>294.21</td>
<td>0</td>
</tr>
<tr>
<td>Carbon intensity source</td>
<td>0</td>
<td>?</td>
<td>2018 electricityMap</td>
<td>EU-28 in 2017</td>
<td>0</td>
</tr>
</tbody>
</table>
Comparison of all selected tools

<table>
<thead>
<tr>
<th>Carbon intensity (gCO2/kWh)</th>
<th>GA</th>
<th>MCI</th>
<th>W</th>
<th>CC</th>
<th>EIT</th>
<th>CT</th>
<th>ES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Carbon intensity source</td>
<td>GA</td>
<td>MCI</td>
<td>W</td>
<td>CC</td>
<td>EIT</td>
<td>CT</td>
<td>ES</td>
</tr>
<tr>
<td>PUE</td>
<td>1.67</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>1.58</td>
<td>1.58</td>
<td>X</td>
</tr>
</tbody>
</table>

Table 4: Energy and carbon emission parameter values
Evaluation

- If I had run this during 24h
  - Energy = 116.82 kWh
    You could boil more than 1000L of water with this energy.

- Carbon footprint = 4 550 gCO2 eq
  Driving 26 km by car emits the same carbon emissions.
Conclusion

• Various tools to compute energy consumption of training an AI model

• Provide it when publishing your model!

• Limits
  • Life Cycle Assessment
  • Cost of production and transport
  • Inference cost and the usage of the model
How to reduce this impact

• Be careful of which Data Center (PUE) you use, and of the location of it (carbon intensity) [GA][CC][EIT][CT]

• Algorithm optimization (random search, memory) [GA][CC][CT]

• Report energy metric and make energy efficient configurations more accessible [CC][EIT]

• Run your algorithms when carbon intensity is low [CC][CT]

• Look for models with high efficiency, not only high accuracy

• Adapt your algorithm and the settings to the hardware [CC][CT]

• Use pretrained models [CC]

• Offset emissions [GA]
References

• Ecoinfo: https://ecoinfo.cnrs.fr/


• GREEN ALGORITHMS
  • https://green-algorithms.org/?
    runTime_hour=24&runTime_min=0&locationContinent=Europe&locationCountry=France&locationRegion=FR&PUEradio=Yes&PUE=1.67&coreType=Both&numberCPUs=40&CPUmodel=Xeon%20E5-2697%20v4&usageCPUradio=Yes&usageCPU=1&numberGPUs=8&GPUmodel=NVIDIA%20Tesla%20V100&usageGPUriad=Yes&usageGPU=1&memory=512&platformType=localServer&PSFradio=Yes&PSF=1

• ML CO2 IMPACT: https://mlco2.github.io/impact/?#compute

• CODECARBON:
  • https://github.com/mlco2/codecarbon
  • medium article: https://medium.com/bcggamma/ai-computing-emits-co%20%E2%82%82%E2%82%82-we-started-measuring-how-much-807dec8c35e3
References

• CARBONTRACKER:
  • https://github.com/lfwa/carbontracker,

• EXPERIMENT IMPACT TRACKER
  • https://github.com/Breakend/experiment-impact-tracker

• ENERGY SCOPE:
  • https://sed-bso.gitlabpages.inria.fr/datacenter/energy_scope.html (to request code)
  • http://energy-scope.bordeaux.inria.fr/ (INRIA server)

• Sources for carbon intensity:
  • https://www.epa.gov/egrid/egrid-summary-tables
  • https://electricitymap.org/
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Developement</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Year</td>
<td>2021</td>
<td>2019</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2020</td>
</tr>
<tr>
<td>By who</td>
<td>UK researchers</td>
<td>Canadian researchers (MILA)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Canadian researchers (MILA)</td>
</tr>
<tr>
<td>Technical</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Availability of source code</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Online availability</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Easiness of installation and use</td>
<td>No install needed</td>
<td>No install needed</td>
<td>Poor</td>
<td>Quite good</td>
<td>Quite good</td>
<td>Good</td>
<td>Fair</td>
</tr>
<tr>
<td>Quality of documentation</td>
<td>Fair</td>
<td>Fair</td>
<td>Good</td>
<td>Fair</td>
<td>Fair</td>
<td>Good</td>
<td>Fair</td>
</tr>
<tr>
<td>Output formats</td>
<td>Graphs and comparisons</td>
<td>Texts and Latex code</td>
<td>JSON</td>
<td>CSV file - Visualisation with Comet</td>
<td>JSON file - Code interface</td>
<td>Logs - Code interface</td>
<td>Online server - Jupyter Notebook - JSON files</td>
</tr>
<tr>
<td>Configuration</td>
<td>Provide carbon emissions</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Carbon intensity source</td>
<td>Local, from electricityMap (2020 value)</td>
<td>Local, listed, including electricitymap</td>
<td>Local, global default value (didn’t find source)</td>
<td>Local, from electricityMap (2018 value)</td>
<td>Various sources, European default value (EU-28 2017)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Carbon intensity value for my experiments (gCO2eq/kWh)</td>
<td>39</td>
<td>432</td>
<td>424.6</td>
<td>47.6</td>
<td>294.21</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Possible manual configuration of carbon intensity</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes but not easy</td>
<td>No</td>
<td></td>
<td></td>
</tr>
<tr>
<td>External factors taken into account (PUE, PSF)</td>
<td>PUE, PSF</td>
<td>Only for cloud providers</td>
<td>None</td>
<td>PUE</td>
<td>PUE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Possible manual configuration of external factors</td>
<td>Yes, default value being PUE = 1.07 (2019), PSF = 1</td>
<td>No</td>
<td>No</td>
<td>Yes, default value PUE = 1.58</td>
<td>No, default value PUE = 1.67</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type of equipment</td>
<td>PC, server, Cloud</td>
<td>PC, server, Cloud</td>
<td>PC, server, Cloud</td>
<td>PC, server, Cloud</td>
<td>PC, server, Cloud</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Other</td>
<td>conversion to other metrics</td>
<td>conversion to other metrics</td>
<td>Interval between measures is configurable, conversion to other metrics, if tracking tools are unavailable will use TDPs from detected hardware, if hardware unknown will provide a default value</td>
<td>Interval between measures is configurable, predictions after one epoch, conversion to other metrics</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Energy model</td>
<td>Hardware taken into account</td>
<td>CPUs, GPUs, Memory</td>
<td>CPUs, GPUs (less hardwares than GA)</td>
<td>Whole machine</td>
<td>CPUs, GPUs, Memory</td>
<td>CPUs, GPUs, Memory</td>
<td>CPUs, GPUs, Memory</td>
</tr>
<tr>
<td>Model used</td>
<td>maximal usage power</td>
<td>maximal usage power</td>
<td>Measure</td>
<td>RAPL files - NVIDIA SMI</td>
<td>RAPL files - NVIDIA SMI</td>
<td>RAPL files - NVIDIA SMI</td>
<td>RAPL files - NVIDIA SMI</td>
</tr>
<tr>
<td>Estimation or Measure</td>
<td>Estimation from maximal usage power</td>
<td>Estimation from maximal usage power</td>
<td>Measure</td>
<td>Estimation from performance counters</td>
<td>Estimation from performance counters</td>
<td>Estimation from performance counters</td>
<td>Estimation from performance counters</td>
</tr>
<tr>
<td>Real time</td>
<td>No</td>
<td>No</td>
<td>25</td>
<td>Yes</td>
<td>No</td>
<td>Per epoch</td>
<td>Yes</td>
</tr>
<tr>
<td>Software</td>
<td>Build-In</td>
<td>CLI</td>
<td>CSV export</td>
<td>Middleware</td>
<td>Online</td>
<td>Carbon emissions</td>
<td>Compatibility</td>
</tr>
<tr>
<td>-----------</td>
<td>----------</td>
<td>-----</td>
<td>------------</td>
<td>------------</td>
<td>--------</td>
<td>------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>RAPL files</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td>✔</td>
<td></td>
<td>Intel CPU</td>
</tr>
<tr>
<td>Nvidia-smi</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td>Nvidia GPU</td>
</tr>
<tr>
<td>sudometrics</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td>MacOS</td>
</tr>
<tr>
<td>Power Gadget</td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td>Intel CPU</td>
</tr>
<tr>
<td>PowDroid</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td>Android</td>
</tr>
<tr>
<td>PowerJoular</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td>Intel CPU</td>
</tr>
<tr>
<td>Greenspector</td>
<td></td>
<td></td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td>Intel CPU</td>
</tr>
<tr>
<td>Scaphandre</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td>Intel CPU</td>
</tr>
<tr>
<td>Power API</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td>Intel CPU</td>
</tr>
<tr>
<td>Energy scope</td>
<td></td>
<td></td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td>Intel CPU, Nvidia GPU</td>
</tr>
<tr>
<td>CodeCarbon</td>
<td>✔</td>
<td>✔</td>
<td>○</td>
<td>Python</td>
<td>✔</td>
<td></td>
<td>Intel CPU, Nvidia GPU</td>
</tr>
<tr>
<td>CarbonTracker</td>
<td></td>
<td></td>
<td>✔</td>
<td>Python</td>
<td>✔</td>
<td></td>
<td>Intel CPU, Nvidia GPU</td>
</tr>
<tr>
<td>Experiment-impact-tracker</td>
<td></td>
<td></td>
<td>✔</td>
<td>Python</td>
<td>✔</td>
<td></td>
<td>Intel CPU, Nvidia GPU</td>
</tr>
<tr>
<td>JoularJX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Java</td>
</tr>
<tr>
<td>Carbonanalyser</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Firefox</td>
</tr>
<tr>
<td>Green algorithms</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td>Most CPU/GPU</td>
</tr>
<tr>
<td>ML CO2 impact</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td>Most CPU/GPU</td>
</tr>
<tr>
<td>Wattmetres</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>