Danmarks Tekniske Universitet

DTU

Quantitative methods to assess sustainability (Task 3)

Authors

Albert Casajuana - s260170
Carlos Lucero - s260219
Jan Metela - s260377
Vlad Burlacu - s225195

May 11, 2026

1 Stakeholder-Oriented Summary and Infographic

Executive Summary

The rapid integration of Artificial Intelligence, specifically Language Models, into global digital infrastructure presents pressing sustainability challenges across environmental, economic, and social dimensions. Based on a comprehensive life-cycle assessment of a proxy Small Language Model (SLM), this summary outlines key insights to guide more sustainable AI development and deployment strategies.

Key Sustainability Insights

The environmental and economic footprints of language models are inextricably linked to energy consumption and compute time. During isolated runs, the training phase dominates resource use; SLM training emissions were measured at magnitudes higher than a single inference (prompting) event. However, the true sustainability challenge emerges at the aggregate level. Millions of concurrent deployments and inference requests compound these impacts, translating local compute tasks into global health risks. These risks disproportionately affect communities situated near energy generation facilities and hardware mining sites.

Influential Parameters and Trade-offs

Architectural complexity and training duration are the primary drivers of total impact. Our assessment reveals a direct relationship: scaling up embedding dimensions or increasing training iterations proportionately increases both embedded CO_2 emissions and cloud-based computing costs. Consequently, stakeholders face a fundamental trade-off: pursuing absolute maximum model performance requires tangible increases in environmental degradation and economic expenditure.

High-Impact Actions for Decision-Makers

- **Right-Sizing Architectures:** Prioritize the deployment of compact, specialized models tailored to specific tasks rather than defaulting to resource-intensive large-scale architectures.
- **Grid-Aware Computing:** Schedule heavy training workloads in data centers powered by low-carbon electricity mixes (e.g., wind and solar) to drastically reduce the CO_2 equivalent per kWh.
- **Efficiency in Development:** Limit unnecessary training cycles by utilizing early stopping protocols and optimizing hyperparameter configurations, preventing excess energy use for diminishing performance returns.

Infographic: SLM Sustainability at a Glance

Training vs. Inference

- **Training**: 0.15 g CO_2e
- **Inference**: 0.0000033 g CO_2e

Training = 45,000× Inference

Safe Operating Space

- **Boundary**: 330 kg CO_2e
- **SLM max**: 0.0003 kg
- **Usage**: <0.0001%

Scale: 1M models = 300 kg

Parameter Impacts

Scenario	Emissions
Baseline	0.15 g
+Iterations	0.30 g (+100%)
+Model size	0.27 g (+80%)
+Batch size	0.29 g (+93%)

Key Actions

1. Optimize architecture first
2. Deploy in low-carbon grids
3. Right-size for tasks
4. Monitor fleet impact

Individual: negligible | Fleet: substantial

2 SLM as a proxy for LLMs

There are several areas in which the analysis of SLMs fails to capture the full complexity and environmental impact of LLMs. In addition to the limitations already defined by the system boundaries, further differences arise due to differences in scale, infrastructure, and deployment. Some of these are discussed below.

2.1 Training

2.1.1 Infrastructure complexity

The SLM used in this analysis was trained on a single personal laptop. In contrast, LLMs require significantly more extensive infrastructure, including large-scale data centers, dedicated facilities for development teams, and in some cases, expansions of electrical grids and power supply systems to meet their energy demands. LLM also requires multiple GPUs to be fast enough, while the SLM was running on single GPU,

This infrastructure must be constructed, maintained, and eventually decommissioned, contributing to additional environmental impacts, economic costs, and potential social implications [4].

2.1.2 Development stage

The life cycle of LLMs includes an extensive development phase, during which models are iteratively improved. This involves experimenting with new architectures, datasets, training pipelines, and hyperparameter configurations.

As a result, multiple versions of the model are trained repeatedly, leading to a substantial increase in cumulative energy consumption and associated emissions. [3]

2.2 Prompting

2.2.1 Model deployment

For LLMs to be economically viable, they must be accessible to users at scale. This requires continuous deployment and maintenance, even during periods of low demand.

Such deployment involves not only running the model on servers but also maintaining supporting infrastructure, including backend systems, user interfaces, and monitoring services. Additionally, there are indirect costs and impacts associated with staffing, maintenance, and operational management.

These factors introduce a significant overhead that is not captured when analyzing SLM-based inference in isolation.

3 Comparison with Production-Scale LLMs

With the SLM having approximately 6.5 million parameters and Mistral Large 2 having 123 billion, it could be estimated that the LLM produces 20,000 times more CO_2 emissions.

Using data obtained from EcoLogits for a production-scale model, generating an output approximately comparable in length to the output size used in this analysis requires about 31.3 mWh of energy with GWP of 2.33 mgCO2eq.

This is roughly half of the energy consumption measured for the SLM (55 mWH). When applying the Danish electricity mix from Task 2, this corresponds to approximately 2×10^{-6} kg of CO_2 emissions per generation.

The emissions are significantly lower than what would be expected if they scaled linearly with parameter count. The EcoLogits data are estimated using life stages similar to our analysis, as they only include the electricity consumed during the prompting (inference) phase [2]. Furthermore, their model accounts for the use of multiple GPUs and Power Usage Effectiveness (PUE) to include data center cooling equipment—factors that were not included in our initial analysis.

While the EcoLogits estimation methodology often that electricity consumption scales with

parameter count, it is important to note that the energy cost depends primarily on the number of active parameters, which can be significantly lower than the total parameter count. [2] Additionally, the discrepancy likely are cause by the use of highly efficient, enterprise-grade GPUs which are capable of handling large batches of requests simultaneously, meaning only a small fraction of their total power draw is dedicated to a single prompt [1].

Finally, it should be noted that the EcoLogits estimation only covers electricity consumed during the operational phase of the server. Similarly to our analysis, it omits critical lifecycle stages such as the construction of the data centers, the manufacturing of the hardware, the power consumed by user devices making the requests, and the necessary networking infrastructure. Consequently, the total environmental impact is likely higher.

4 Rebound Effect

Technological advancements can make large language models (LLMs) more energy efficient, but this does not necessarily reduce total CO emissions. Lower training and inference costs make LLMs more accessible, directly interacting with a rapidly growing global demand for AI services. As these models become cheaper to operate, they are seamlessly integrated into more daily applications, increasing the total volume of inference requests to a point that often outweighs the initial per-prompt energy savings.

Likewise, efficiency improvements, such as achieving similar performance with fewer parameters, do not always lead to smaller deployed models. Instead, companies often reinvest these gains into larger or more powerful models, prioritizing performance capabilities and market dominance over absolute reductions in resource consumption.

5 Circular Economy

To reduce the environmental impact of large language models, future development could follow the principles of narrowing, slowing, and closing the resource loop:

Narrowing: Focus on more efficient and specialized architectures instead of massive general-purpose models. Task-specific models require fewer parameters and less training energy.

Slowing: Extend the lifespan of trained models by releasing older pretrained weights and models as open-source. This allows communities to continue using and fine-tuning existing models, preserving the energy invested in training.

Closing: Improve recycling of hardware by recovering rare earth metals from decommissioned GPUs and servers.

Appendix: Tables and References

References

[1] Julien Delavande, Regis Pierrard, and Sasha Luccioni. Understanding efficiency: Quantization, batching, and serving strategies in llm energy use, 2026.

[2] EcoLogits. Llm inference methodology: Assumptions and limitations. https://ecologits.ai/latest/methodology/llm_inference/#assumptions-and-limitations, 2026. Accessed: 2026-05-10.

[3] Jacob Morrison, Noah A. Smith, and Emma Strubell. The hidden cost of thinking: Energy use and environmental impact of lms beyond pretraining, 2026.

[4] Rachel Mural, Dipesh Pherwani, Chaitanya Gupta, Yiqi Yu, Ai Takahashi, Dongjoo Kim, Subir Majumder, Henry Lee, Minlan Yu, and Le Xie. Ai, data centers, and the u.s. electric grid: A watershed moment. Technical report, Belfer Center for Science and International Affairs, February 2026. Accessed: 2026-05-10.

Declaration of Generative AI Use

The authors declare that Claude (Anthropic) was used to assist with literature review, text editing, code debugging, and LaTeX formatting. All technical analysis, data collection, experimental design, and interpretation of results were conducted by the authors. AI-generated content was reviewed and verified for accuracy. The authors take full responsibility for the content of this report.

Danmarks Tekniske Universitet

DTU

Quantitative methods to assess sustainability (Task 2)

Authors

Albert Casajuana - s260170
Carlos Lucero - s260219
Jan Metela - s260377
Vlad Burlacu - s225195

May 11, 2026

1 Environmental impacts

1.1 Data and Carbon Footprint

Energy consumption during training and inference was measured using CodeCarbon on an NVIDIA GeForce RTX 3060 Laptop GPU (rated 60–115 W [3]), consistent with the measured values in Table 1. The electricity mix at the time of measurement (Table 2) was obtained from publicly available sources [2], resulting into a weighted average emission intensity of approximately 0.059 kg CO_2e/kWh, dominated by wind and low-carbon imports.

The resulting CO_2 emissions are reported in Table 3. S1 and S3 shows linear relationship with the amount of data used during training and the produced emissions. S2 proves that scaling the model parameters to increase the performance increases the environmental impact.

Prompting emissions are two orders of magnitude lower than training across all scenarios. Increasing the number of generated tokens (S5) leads to higher emissions, while increasing prompt length S4 has only a minor effect. This shows that the training phase dominates the lifecycle environmental impact of the SLM.

1.2 Safe Operating Space

We apply the Grandfathering (GF) allocation approach, where the allocated environmental budget is proportional to the activity's current share of global emissions [5]. Under the 1.5°C Paris Agreement target, the global carbon budget is approximately 33 Gt CO_2/year [6]. Attributing 1% to AI operations and assuming 10^6 global training runs per year [4], the allocated boundary per training run is 330 kg CO_2. All three scenarios fall several orders of magnitude below this boundary (Table 4). The boundary is sensitive to the assumed number of global training runs: using 10^7 tightens it to 33 kg CO_2e, but all scenarios remain well within the safe operating space. These results reflect the small scale of the SLM and the low CO_2 electricity mix; the sustainability challenge of AI lies in the aggregate impact of millions of concurrent deployments, and scale of LLMs.

2 Economic impacts

2.1 Life Cycle Costing

To estimate the economic cost of training, we use AWS EC2 `g4dn.xlarge` instances (NVIDIA T4 GPU) as a reference cloud environment, priced at $0.526/hour [1]. Training times were taken from Table 1. The main cost components are compute time, with energy cost embedded in the cloud rate, and a negligible storage cost for the resulting model checkpoint. Results are shown in Table 5.

Compute time dominates total cost, scaling with training duration and model size. S1 costs approximately 2× more than base scenario, proportional to the doubling of training

iterations. These cost ratios mirror the CO_2 ratios in Table 3, confirming that at this scale, environmental and economic impacts are tightly coupled through energy consumption. At production scale (e.g. GPT-3 at $4.6M estimated training cost [4]), compute cost becomes the dominant driver of both economic and environmental impact, reinforcing the importance of architectural efficiency.

3 Social impacts and human health

We adopt the perspective of a large-scale fleet of SLMs deployed across organisations, as this better reflects the aggregate societal implications of the technology than a single isolated model.

The primary stakeholder groups affected are: *data centre workers and local communities*, exposed to heat, noise, and water consumption from cooling infrastructure; *populations in electricity-producing regions*, particularly those dependent on fossil-fuel grids, who bear the health burden of particulate matter and GHG emissions from energy generation; and *end users and society broadly*, affected by privacy risks from training data, algorithmic bias embedded in model outputs, and the concentration of AI infrastructure in a small number of organisations.

Main human health risks include respiratory and cardiovascular impacts from energy-related air pollution, particularly in regions with coal-heavy electricity mixes. Another risks are associated with mining materials for computer hardware (like rare earth metals and semiconductors). Those materials are often mined in poorer countries. This means the people who use AI don't usually face the health risks, but others do [7]. Privacy risks from large-scale data collection used in training represent an additional social harm.

Applying an impact factor of 4.0×10^{-7} DALY/kWh, the baseline scenario S0 consumes 0.002587 kWh and yields 1.03×10^{-9} DALYs, whereas S1 consumes 0.005083 kWh and yields 2.03×10^{-9} DALYs. In combination with the higher emissions intensity associated with the less efficient architecture, these results indicate that scaling such architectural inefficiencies compounds global health harm for communities located near power generation facilities.

Appendix: Tables and References

References

[1] Amazon Web Services. Amazon EC2 on-demand pricing, 2026. Accessed: 2026-04-18.

[2] Electricity Maps. Electricity maps dk2 live 15-minute data. https://app.electricitymaps.com/map/zone/DK-DK2/live/fifteen_minutes, 2026. Accessed: 2026-04-04.

[3] NVIDIA. Geforce rtx 30 series laptops specs, 2026. Accessed: 2026-04-04.

[4] David Patterson, Joseph Gonzalez, Quoc Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David So, Maud Texier, and Jeff Dean. Carbon emissions and large neural network training, 2021.

[5] Johan Rockström, Will Steffen, Kevin Noone, et al. A safe operating space for humanity. *Nature*, 461:472–475, 2009.

[6] United Nations Environment Programme. Emissions gap report 2022. Technical report, UNEP, 2022.

[7] Qiang Zhang, Xujia Jiang, Dan Tong, Steven J. Davis, et al. Transboundary health impacts of transported global air pollution and international trade. *Nature*, 543:705–709, 2017.

Tables

Scenario	Changed parameter	Value	Duration (s)	Power (W)	Energy (kWh)
base training	-	-	82.63	98.26	0.002587
S1-train	Training iterations	4000	162.93	97.46	0.005083
S2-train	Embedding size	256	144.05	99.27	0.004553
S3-train	Batch size	64	152.72	98.44	0.004850
base-prompt	-	-	4.39	32.04	0.000055
S4-prompt	Prompt length	41 char	4.35	26.70	0.000049
S5-prompt	Generated tokens	400	6.60	33.42	0.000087

Table 1: Training and Prompting Energy and Power Consumption.

Electricity source	Share (%)	CO_2 intensity (kg/kWh)
Biomass	15.65	0.230
Solar	16.33	0.036
Wind	34.77	0.013
Import from Sweden	21.81	0.024
Import from Germany	11.35	0.067

Table 2: Electricity mix during training and prompting (DK2 zone).

Scenario	Energy (kWh)	CO_2e (kg)
base-train	0.002587	0.00015
S1-train	0.005083	0.00030
S2-train	0.004553	0.00027
S3-train	0.004850	0.00029
base-prompting	0.000055	3.26×10^{-6}
S4-prompting	0.000049	2.90×10^{-6}
S5-prompting	0.000087	5.15×10^{-6}

Table 3: Estimated CO_2e emissions per scenario.

Scenario Name	Emissions (kg)	Boundary (kg)	Ratio (Emissions/Boundary)
base-train	0.00015	330	4.55×10^{-7}
S1-train	0.00030	330	9.09×10^{-7}
S2-train	0.00027	330	8.18×10^{-7}
S3-train	0.00029	330	8.79×10^{-7}
base-prompting	3.26×10^{-6}	330	9.88×10^{-9}
S4-prompting	2.90×10^{-6}	330	8.79×10^{-9}
S5-prompting	5.15×10^{-6}	330	1.56×10^{-8}

Table 4: Emissions comparison with fixed boundary of 330 kg

Scenario	Duration (s)	Duration (h)	Cost ($)
base-train	82.63	0.02295	0.0121
S1-train	162.93	0.04526	0.0238
S2-train	144.05	0.04001	0.0210
S3-train	152.72	0.04242	0.0223
base-prompt	4.39	0.00122	0.00064
S4-prompt	4.35	0.00121	0.00064
S5-prompt	6.60	0.00183	0.00096

Table 5: Training and prompting duration and estimated cost at $0.526/hour

Danmarks Tekniske Universitet

DTU

Quantitative methods to assess sustainability (Task 1)

Authors

Albert Casajuana - s260170
Carlos Lucero - s260219
Jan Metela - s260377
Vlad Burlacu - s225195

May 11, 2026

1 Context and motivation

Large Language Models (LLMs) offer strong abilities in natural language processing and generation, with use-cases ranging from education and workplace efficiency to advanced human–machine interactions. Consequently, these models are widely used by both individuals and organizations [1]. The scale of this usage is evident on platforms such as OpenAI's ChatGPT, which processes approximately 2.5 billion user prompts daily [2].

Training LLMs involves processing of large datasets over multiple epochs, a resource-intensive task requiring significant amounts of electricity and water, alongside specialized high-performance computing (HPC) infrastructure [3].

Global sustainability efforts aim to mitigate climate change driven by rising CO2 emissions [4]. Given that the lifecycle of LLMs contributes to global energy demand, quantifying the environmental footprint of these services is necessary to understand and manage their impact on the environment.

2 Object of assessment

The object of assessment is a Small Language Model (SLM) designed for character-level language modeling. The architecture is a simplified variant of the GPT-style Transformer, originally proposed by Vaswani et al. [5] for machine translation tasks.

In accordance with Life Cycle Assessment (LCA) methodology, we define the **functional units** (FUs) to provide a quantified reference for the service performance of the system [6]. The focus is on operational phase of the model, specifically targeting the energy-intensive stages of training and deployment. A detailed definition of the system boundaries and the justification for specific stage exclusions are provided in Section 3.

The following FUs are established for the assessment:

- **Training:** Training a SLM model with decoder-only transformer architecture on the *Tiny Shakespeare* dataset containing 1,115,394 characters using a danish electricity mix and RTX 3060 laptop GPU.
- **Inference FU:** Generation of a response to a prompt using SLM decoder-only transformer model trained on Tiny Shakespeare dataset run on RTX 3060 laptop GPU powered by a danish electricity mix.

3 System boundaries

To keep the analysis manageable within the scope of this assignment, only the primary source of energy consumption (GPU computation during training and prompting) is considered.

Secondary factors with high variability or measurement complexity are excluded.

To enable sensitivity analysis for both Life Cycle Assessment (LCA) and Life Cycle Costing (LCC), the operational phase is divided into a training phase and a prompting phase.

3.1 Training

Included:

- The total energy consumed by the GPU during the model training run and the associated CO_2 emissions.

Excluded:

- Hyperparameter optimization, as different hyperparameter configurations are intended for different scenarios.
- Emissions associated with preparing and preprocessing the dataset, as this is a highly time-consuming and variable task that is difficult to estimate or simulate.
- The process of evaluating and refining the model output to be user friendly, as this requires extensive user interaction and feedback collection, which is beyond the scope of this analysis.

3.2 Prompting

Included:

- The total energy consumed by the GPU during input processing and output generation, along with the associated CO_2 emissions.

3.3 Shared

The following lifecycle stages are relevant for both phases. They were excluded due to their high complexity, which would increase the difficulty of conducting the analysis.

- Extraction of raw materials required for the hardware and the manufacturing of the hardware.
- Decommissioning of the hardware.

4 Scenarios

To support a sensitivity analysis, we use one baseline configuration and five single-parameter scenarios. In each scenario, only one parameter is varied across a defined range, while all

other settings remain fixed.

Baseline configuration

The baseline is the default model and prompt setup used as the reference point for all comparisons.

- **Architecture:** N_LAYER = 4, N_HEAD = 4, N_EMBD = 128, Dropout = 0.1, Bias = True.
- **Training:** Batch size = 32, Block size = 256, MAX_ITERS = 2000, Learning rate = 3e-4, Weight decay = 0.1, Grad clip = 1.0, DTYPE = float32.
- **Hardware:** Device = RTX 3060 laptop GPU.
- **Prompting:** Prompt "To, be or not to be" (19 chars) and a maximum response length of 200 tokens.

Training scenarios

The three training scenarios are selected because they represent common choices that directly affect computational work during model training.

S1 – Number of training iterations Increases of MAX_ITERS, included because training duration has a direct effect on total compute time and electricity use. The aim is to assess how much the additional training required to achieve better performance influences the power demand.

S2 – Embedding dimension Increases of N_EMBD, included because embedding size strongly affects model width, memory use, and the amount of computation per forward and backward pass. The aim is to assess how increasing the model size changes the power demand.

S3 – Batch size Increases batch size, because batch size changes how much data is processed per training step and can affect both runtime efficiency and memory demand. The aim is to assess how larger or smaller batches change electricity demand.

Prompt scenarios

The two prompt scenarios are selected to represent common modification required to answer the queries.

S4 – Input prompt length Increases the length of the input prompt while keeping the response setting fixed. It is included because longer prompts require more tokens to be processed before generation starts.

S5 – Maximum response length Increases the maximum number of generated tokens. It is included because generating more tokens increases inference time and is required to answer more demanding queries.

References

[1] W. Liang, Y. Zhang, M. Codreanu, J. Wang, H. Cao, and J. Zou, "The widespread adoption of large language model-assisted writing across society," *Patterns (New York, N.Y.)*, vol. 6, no. 12, p. 101366, 2025.

[2] ETtech, "Chatgpt now handles 2.5 billion user prompts daily," July 2025. The Economic Times. Last updated July 22, 2025.

[3] Z. Ji and M. Jiang, "A systematic review of electricity demand for large language models: evaluations, challenges, and solutions," *Renewable and Sustainable Energy Reviews*, vol. 225, p. 116159, 2026.

[4] W. Steffen, J. Rockström, K. Richardson, T. M. Lenton, C. Folke, D. Liverman, C. P. Summerhayes, A. D. Barnosky, S. E. Cornell, M. Crucifix, J. F. Donges, I. Fetzer, S. J. Lade, M. Scheffer, R. Winkelmann, and H. J. Schellnhuber, "Trajectories of the earth system in the anthropocene," *Proceedings of the National Academy of Sciences*, vol. 115, no. 33, 2018.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.

[6] B. P. Weidema, "The product, functional unit and reference flows in lca," 2004. Guideline from the Danish Environmental Protection Agency.