Green Artificial Intelligence: Optimising the Energy Consumption and Carbon Footprint of Large Language Models

Srijib Samanta
Assistant Professor, Position: Head of Department (HOD) Department: Bachelor of Computer Application (BCA), Tamralipta Institute of Management & Technology

***Abstract*—Large language models (LLMs) have created new capabilities in language generation, reasoning, coding and knowledge access, but their environmental cost extends across experimentation, training, deployment, data-centre overhead and hardware manufacture. This paper examines how Green Artificial Intelligence can reduce energy consumption and carbon emissions without treating model quality as the only measure of progress. A structured narrative review and lifecycle analysis are used to synthesise evidence on efficient architectures, sparsity, quantisation, distillation, retrieval-augmented generation, hardware-aware serving, workload scheduling and carbon accounting. The paper proposes a Green LLM Optimisation Framework built around five actions: measure the baseline, match model capacity to task difficulty, optimise the compute stack, shift flexible workloads to lower-carbon times and locations, and report quality–energy–carbon trade-offs. The analysis shows that a single universal "energy per prompt" value is misleading because energy depends on model architecture, precision, prompt and output length, batch size, accelerator utilisation, cooling overhead and electricity-grid intensity. Training receives public attention, yet repeated inference can dominate operational impact when services run at scale. Consequently, sustainable LLM deployment requires optimisation at the level of tokens, requests, models, servers and data centres. Green AI should therefore be understood not as a restriction on innovation but as a discipline for producing useful intelligence with the least defensible lifecycle burden. The paper concludes with a practical measurement protocol, governance recommendations and a research agenda for transparent, comparable and carbon-aware LLM systems.**

***Index Terms*—Green AI; large language models; energy efficiency; carbon footprint; sustainable computing; quantisation; carbon-aware scheduling; inference optimisation.**

I. INTRODUCTION:

The rapid diffusion of generative AI has shifted computational sustainability from a specialist concern to a strategic research issue. LLMs consume electricity while researchers search for architectures and hyperparameters, during pre-training and fine-tuning, and whenever users request an output. Their infrastructure also requires networking, storage, cooling and the manufacture and replacement of accelerators.

The International Energy Agency reported that data centres consumed around 1.5% of global electricity in 2024 and projects their electricity use to approach 945 TWh by 2030 in its base case; AI is not the sole cause, but accelerated servers are a major source of growth (IEA, 2025).

Green AI calls for efficiency to be treated as a research outcome alongside accuracy (Schwartz et al., 2020). This perspective is especially important for LLMs because parameter count is an incomplete proxy for environmental impact.

A smaller model used inefficiently can waste more energy than a larger model operated at high utilisation, and a low-energy workload can still have high operational emissions when electricity is carbon intensive.

This paper therefore separates energy, operational carbon and embodied carbon and asks how the complete LLM lifecycle can be optimised.

II. RESEARCH CONTEXT AND LITERATURE REVIEW:

2.1. From "Red AI" to Green AI

Schwartz et al. (2020) distinguished efficiency-centred Green AI from a "Red AI" culture in which marginal accuracy gains are pursued through disproportionately larger computational budgets. The argument is not that large models are inherently unacceptable. It is that claims of progress should disclose the resources required and compare gains against compute, energy and cost. This improves reproducibility and makes AI research more accessible to institutions that do not control hyperscale infrastructure.

2.2. Early Carbon Accounting for Machine Learning

Strubell et al. (2019) made the environmental cost of computationally intensive natural-language processing visible, particularly where neural architecture search multiplies training runs. Henderson et al. (2020) subsequently proposed systematic reporting of energy and carbon, noting that comparisons become unreliable when hardware, location and accounting boundaries are omitted. Patterson et al. (2021) demonstrated that algorithm, processor, data-centre efficiency and electricity supply can jointly produce orders-of-magnitude differences in emissions.

Their findings also show why historic carbon estimates should not be transferred mechanically to a different model or infrastructure.

2.3. Evidence from Large Language Models

Luccioni et al. (2023) conducted a lifecycle-oriented assessment of BLOOM, separating the dynamic electricity used in final training from wider processes that include equipment manufacture and operational overhead.

The study estimated approximately 24.7 tonnes of CO_2-equivalent for dynamic training electricity and 50.5 tonnes under the broader boundary. Its significance lies less in a number that can be generalised to all LLMs than in its demonstration that the selected system boundary changes the result. Faiz et al. (2024) extended this logic through LLMCarbon, which models operational and embodied carbon across training, inference, experimentation and storage for dense and mixture-of-experts architectures.

2.4. The Growing Importance of Inference

Training is a concentrated event; inference is repeated for the life of a deployed service. Energy per request depends on input processing, autoregressive output generation, memory traffic, batching and hardware utilisation. Recent benchmarking frameworks consequently analyse energy, water and carbon per prompt or token and emphasise infrastructure-aware comparisons (Jegham et al., 2025). At high request volumes, apparently small per-query impacts accumulate. However, public estimates for proprietary systems often infer hardware and operational conditions, so they should be reported with uncertainty rather than as exact measurements.

2.5. Research Gap

The literature supplies valuable measurement tools and isolated optimisation techniques, but three gaps remain. First, energy-reduction methods are often assessed separately even though deployment decisions interact. Second, papers frequently report power or energy without converting it through transparent data-centre and grid assumptions. Third, accuracy-only benchmarking does not reveal whether a model is oversized for the intended task. A compact lifecycle framework is therefore needed to connect task selection, model design, serving, carbon-aware operations and reporting.

III. OBJECTIVES AND RESEARCH METHODOLOGY

3.1. Objectives

- To identify the principal sources of energy use and carbon emissions across the LLM lifecycle.
- To critically evaluate technical and operational strategies for reducing LLM energy demand.
- To distinguish energy efficiency from carbon efficiency and define appropriate measurement boundaries.
- To formulate an integrated Green LLM Optimisation Framework for research and deployment.
- To recommend transparent university, industry and policy reporting practices.

3.2. Research Questions

- Which lifecycle stages and workload characteristics determine the energy and carbon footprint of LLMs?

- Which optimisation techniques can reduce energy while preserving task-appropriate quality?
- How should researchers measure and report LLM environmental performance comparably?
- What institutional practices can align LLM innovation with sustainability?

3.3. Research Design

The paper adopts a structured narrative review combined with conceptual lifecycle analysis. It is not presented as a primary hardware experiment. Peer-reviewed articles, recognised conference papers, technical measurement studies and authoritative energy reports were prioritised. Sources were selected when they addressed at least one of four domains: LLM/ML energy measurement, carbon accounting, model or systems optimisation, and data-centre electricity. Evidence was compared through a common analytical matrix containing lifecycle stage, unit of measurement, system boundary, optimisation lever, reported trade-off and limitation.

3.4. Analytical Boundaries

The analysis distinguishes operational electricity from embodied impact. Operational energy includes computation, memory, storage, networking and the allocated share of cooling and power conversion. Operational carbon is produced by applying a location- and time-sensitive grid-emission factor to electricity consumption. Embodied carbon includes emissions associated with manufacturing, transporting and retiring hardware. Water and land impacts are important but remain outside the primary outcome because comparable disclosures are less mature.

3.5. Core Equations and Metrics

IT energy is estimated by integrating measured equipment power over time:

$$E_{IT}(kWh) = \Sigma[P_{CPU(t)} + P_{GPU(t)} + P_{memory(t)}] \times \frac{\Delta t}{1000}$$

Facility energy incorporates power-usage effectiveness (PUE):

$$E_{facility} = E_{IT} \times PUE$$

Location-based operational emissions are:

$$CO^2e_{operational}(kg) = E_{facility}(kWh) \times CI_{grid}\left(kg\,\frac{CO^2e}{kWh}\right)$$

For meaningful comparison, environmental cost must be normalised by useful service. Recommended indicators include joules per generated token, watt-hours per successful request, kilograms CO^2e per million tokens, energy–delay product, and quality per kilowatt-hour. “Successful request” should be defined through a task-specific quality threshold rather than simple completion.

3.6. Methodological Limitations

Vendor opacity restricts direct measurement of proprietary models. Grid carbon intensity varies by hour and accounting method. PUE is a facility average and may not equal marginal overhead for one job. Allocating embodied emissions requires assumptions about hardware lifetime and utilisation. The framework therefore favours ranges, sensitivity analysis and explicit boundaries over false precision.

IV. WHERE LLM ENERGY AND CARBON ARISE

4.1. Research and Experimentation

Model development can involve data filtering, tokenisation, pilot runs, hyperparameter search, ablation studies and failed experiments. A final training run therefore understates the energy required to produce a model. Experiment tracking should associate every run with hardware type, duration, utilisation, energy and result, enabling low-value runs to be terminated early and previous checkpoints to be reused.

4.2. Pre-training and Fine-tuning

Pre-training processes vast token corpora through repeated matrix operations and distributed communication. Energy is affected by model width and depth, token count, precision, sequence length, parallelisation, checkpoint frequency and accelerator efficiency. Fine-tuning is smaller but can be repeated across many tasks and customers. Parameter-efficient fine-tuning reduces the number of updated weights and can avoid maintaining numerous full model copies.

4.3. Inference

Inference has two computationally different phases. Prefill processes input tokens in parallel and is compute intensive; decoding produces output tokens

sequentially and is often limited by memory movement. Long contexts increase attention and key–value cache requirements, while long outputs keep accelerators active. Request batching can improve utilisation but may increase latency. Sustainable serving must therefore optimise the required quality under explicit latency and throughput constraints.

4.4. Data-Centre Overhead and Electricity Supply

Accelerator energy is only part of facility demand. Cooling, fans, pumps, uninterruptible power supplies and conversion losses are represented approximately through PUE. Carbon intensity depends on the electricity supplying the data centre. Patterson et al. (2021) showed that geography and carbon-free energy availability materially affect emissions. Carbon-aware scheduling can move flexible training or batch inference to a cleaner hour or region, but claims should distinguish location-based grid emissions from contractual market-based claims.

4.5. Embodied Carbon and Rebound

Accelerator manufacture involves semiconductor fabrication, materials, transport and server integration. Replacing hardware solely for better performance can shift environmental pressure from electricity to embodied emissions. Moreover, efficiency can lower cost and increase demand—a rebound effect that partly or fully offsets per-request savings. Organisations should therefore track total monthly energy and emissions in addition to intensity metrics.

Lifecycle stage	Dominant drivers	Preferred measurement	Primary optimisation
Experimentation	Run count, search space, failed jobs	kWh per accepted experiment	Early stopping; reuse; efficient search
Training	Tokens, precision, parallelism, utilisation	kWh/run; CO_2e/run; quality/kWh	Data/model scaling; sparsity; efficient hardware
Fine-tuning	Number of tasks and model copies	kWh/adaptation	LoRA/adapters; shared base model
Inference	Input/output length, batch, latency, cache	Wh/request; J/token; CO_2e/1M tokens	Routing; quantisation; batching; caching
Infrastructure	PUE, grid intensity, embodied equipment	Facility kWh; time/location CO_2e	Carbon-aware placement; longer hardware life

V. TECHNICAL OPTIMISATION OF LLMs

5.1. Right-Sizing and Model Routing

The most direct green AI decision is to avoid using a frontier-scale model for a task that a smaller model can perform reliably. A routing layer can classify request difficulty, send routine work to a compact model and escalate uncertain or complex cases. The routing policy must include a quality safeguard; energy savings achieved by generating unusable answers merely transfer cost to retries and human correction.

5.2. Quantisation

Quantisation represents weights and sometimes activations with fewer bits, reducing memory capacity, bandwidth and potentially computation. Eight-bit and four-bit deployment can make larger models fit on fewer devices. Actual energy gains depend on kernel and hardware support: a compressed representation that is repeatedly converted or executed by an inefficient kernel may save memory without achieving proportional energy reduction. Evaluation must therefore measure wall-plug energy, latency and task quality on the target platform.

5.3. Pruning, Sparsity and Mixture of Experts

Pruning removes low-importance weights or structures. Structured sparsity is generally easier for accelerators to exploit than irregular sparsity. Mixture-of-experts models activate only part of the parameter set for each token, potentially reducing arithmetic relative to a similarly sized dense model. Yet routing, communication, load imbalance and the memory required to host inactive experts can reduce the expected gain. Active parameters and measured energy are more informative than total parameters alone.

5.4. Knowledge Distillation

Distillation trains a smaller student to reproduce selected behaviours of a larger teacher. It transfers capability into a cheaper serving model and is

particularly attractive for stable, high-volume tasks. Its environmental balance should include the one-time teacher-generated data and student training cost, then compare these with inference savings over the expected service volume. Distillation may be unjustified for a short-lived or rarely used application.

5.5. Parameter-Efficient Adaptation

Low-rank adaptation, adapters and prompt tuning modify a limited set of parameters rather than every model weight. These methods reduce training memory, optimiser state and storage for multiple task variants. They also enable a shared base model, although serving many adapters can introduce switching and cache-management overhead that must be measured.

5.6. Retrieval, Caching and Context Control

Retrieval-augmented generation can allow a smaller model to answer domain questions using selected external evidence, but retrieval is not environmentally free. Embedding generation, vector search, reranking and enlarged prompts add computation. Efficient RAG retrieves only what is needed, deduplicates documents and imposes a context budget. Semantic and prefix caching can eliminate repeated prefill work, while output-length limits prevent unnecessary decoding.

5.7. Systems and Hardware Optimisation

- Use fused kernels, optimised attention and memory management appropriate to the accelerator.
- Increase utilisation through dynamic batching while enforcing latency service levels.
- Apply continuous batching and paged key–value caches to reduce fragmentation.
- Select accelerators by measured task-level energy efficiency, not peak theoretical operations alone.
- Consolidate low-volume services and power down genuinely idle resources where operationally safe.
- Profile CPU, GPU, RAM and networking so that savings are not displaced to an unmeasured component.

Technique	Likely benefit	Principal risk
Right-sized routing	Avoids unnecessary high-capacity inference.	Misrouting harms quality or increases retries.
Quantisation	Lower memory and bandwidth; fewer devices.	Quality loss or unsupported low-bit kernels.
Distillation	Low-cost high-volume serving.	Up-front teacher and training cost.
Sparse/MoE	Fewer active computations per token.	Communication and load imbalance.
RAG/caching	Reduces memorisation burden and repeated work.	Retrieval and long-context overhead.
Batching	Higher accelerator utilisation.	Latency and queueing trade-off.

VI. GREEN LLM OPTIMISATION FRAMEWORK

The proposed framework treats sustainability as an iterative engineering constraint. It can be applied during model procurement, training, fine-tuning or deployment.

Stage	Decision question	Required evidence	Output
1. Measure	What is the present energy and carbon baseline?	Power telemetry, duration, tokens, PUE, grid intensity, quality.	Auditable baseline with uncertainty.
2. Match	What is the smallest model that meets the quality and safety threshold?	Task benchmark, failure analysis, latency target.	Model/routing policy.
3. Minimise	Which model and system changes reduce energy without unacceptable loss?	A/B energy, latency, throughput and quality tests.	Optimised compute stack.
4. Move	Can flexible work run at a cleaner time or place?	Hourly/location carbon intensity, data and latency constraints.	Carbon-aware schedule/placement.
5. Monitor	Do savings persist under production traffic?	Total energy, intensity, demand, drift and rebound.	Operational dashboard and alerts.
6. Make public	Can another researcher interpret the claim?	Boundary, hardware, software, workload and factors.	Transparent Green AI statement.

6.1. Quality-Constrained Optimisation

A Green AI system should minimise lifecycle impact subject to requirements for quality, safety, latency, privacy and availability. One conceptual objective is:

$$Minimise\ J = \alpha E + \beta C + \gamma L + \delta K, subject\ to\ Q \geq Qmin\ and\ S \geq Smin$$

Here E is energy, C carbon, L latency, K monetary cost, Q task quality and S safety/reliability. The coefficients express organisational priorities. Carbon is not redundant with energy because the same kWh can have different emissions across time and location. A Pareto frontier should be reported when no single configuration is best on every criterion.

6.2. Practical Experimental Protocol

- Fix a representative prompt set and specify input/output token distributions.
- Warm the system and measure an idle baseline separately.
- Repeat each configuration enough times to quantify variability.
- Record accelerator, CPU and memory power at the highest reliable sampling rate available.
- Report batch size, precision, serving engine, model revision, latency and throughput.
- Apply facility overhead and grid intensity transparently; provide sensitivity ranges.
- Evaluate accuracy, factuality or task success using a predefined threshold.
- Compare total energy as well as per-token and per-successful-request intensity.

6.3. Decision Rule

An optimisation should be adopted when it achieves a statistically and operationally meaningful reduction in facility energy or CO_2e, satisfies the task-quality threshold and does not create a material unreported burden elsewhere. If a low-energy configuration produces more retries, longer human review or increased retrieval, the full workflow rather than the model endpoint must be compared.

6.4. Worked Illustrative Calculation

Assume a monitored inference experiment records 8.0 kWh of IT energy for 20,000 successful requests. With PUE = 1.20, facility energy is 9.6 kWh. If the applicable grid factor is 0.45 kg CO_2e/kWh, operational emissions are 4.32 kg CO_2e, or 0.216 g per successful request. If a quantised configuration lowers IT energy to 5.6 kWh while maintaining the same task-success threshold, facility energy becomes 6.72 kWh and operational emissions 3.02 kg CO_2e. These values are illustrative, not empirical findings; their purpose is to demonstrate transparent calculation and boundary reporting.

VII. DISCUSSION: TRADE-OFFS AND INTERPRETATION

7.1. There Is No Universal Energy-per-Prompt Constant

A prompt is not a standard physical unit. It may contain ten or ten thousand tokens and request a short classification or a long chain of generated text. Model family, precision, hardware, batch size, queue depth, cache reuse and serving engine all alter energy. Comparisons should therefore stratify by workload and report token distributions, latency and quality. Estimates for closed APIs are valuable for awareness but should clearly state when hardware is inferred rather than directly measured.

7.2. Efficiency and Carbon Can Move Differently

Reducing kWh always reduces operational emissions for the same electricity supply, but a cleaner location can lower carbon without lowering energy. Conversely, moving a job may increase data transfer or delay. Carbon-aware scheduling should complement, not substitute for, energy efficiency. Organisations should publish both quantities and avoid presenting renewable-energy certificates as evidence that a workload used no physical electricity or caused no marginal system impact.

7.3. Training versus Inference

The dominant lifecycle stage depends on use. A foundation model that is trained once and rarely deployed may be training dominated. A popular assistant serving continuous traffic can accumulate a larger inference burden. Break-even analysis is essential: a costly optimisation such as distillation is environmentally rational only when future per-request savings exceed its creation cost within the system's expected life.

7.4. Quality, Safety and Inclusion

A smaller or highly compressed model may be less robust for minority languages, difficult reasoning or

safety-critical instructions. Green AI must not externalise environmental savings into discriminatory errors, misinformation or unpaid human correction. Evaluation should include subgroup performance and failure severity. Model routing is preferable to a single small model when difficult cases can be detected and escalated reliably.

7.5. Rebound and Absolute Impact

Efficiency lowers the resource required per unit of service, often lowering price and enabling new uses. Total emissions may still rise when demand expands faster than efficiency improves.

This is why intensity metrics must be paired with absolute monthly electricity, carbon and request volume. A production dashboard should show both the slope of improvement and total environmental pressure.

7.6. Indicative Findings from the Synthesis

- Measurement is the first optimisation: undisclosed boundaries prevent credible comparison.
- Model right-sizing and routing often offer more fundamental savings than low-level tuning alone.
- Inference optimisation is indispensable for high-volume services, even when training has already been completed.
- Hardware utilisation, PUE and grid intensity can materially change the carbon consequence of identical algorithms.
- Compression gains must be verified on target hardware and with task-level quality constraints.
- Operational carbon accounting should be extended toward embodied hardware and full-service workflows.

7.7. Threats to Validity

The review is limited by rapidly changing hardware and model software, uneven disclosure by commercial providers and different carbon-accounting conventions.

Reported results cannot be pooled naively because workloads and boundaries differ. The proposed framework is consequently a decision and reporting structure rather than a claim that one optimisation produces a fixed percentage reduction in every setting.

VIII. GOVERNANCE AND IMPLEMENTATION RECOMMENDATIONS

8.1. For Researchers and Universities

- Include a computational budget in the protocol: device type, estimated hours, experiment ceiling and stopping criteria.
- Report energy and CO_2e alongside task quality for resource-intensive studies.
- Use shared checkpoints and parameter-efficient methods before training new models from scratch.
- Maintain an experiment registry to reduce duplicated failed runs and enable reproducibility.
- Provide students access to measurement tools such as power telemetry and CodeCarbon, while teaching their assumptions and limitations.
- Evaluate research contribution per unit of compute rather than rewarding parameter count as a proxy for novelty.

8.2. For Developers and Cloud Providers

- Expose energy and carbon telemetry through serving and training dashboards.
- Publish model cards containing recommended efficient configurations and measured workload ranges.
- Offer carbon-aware scheduling with clear service-level and data-residency controls.
- Support low-precision kernels, efficient attention, auto-scaling and safe idle-power management.
- Disclose location-based electricity factors, PUE methodology and the treatment of embodied emissions.

8.3. For Organisations Procuring LLM Services

Procurement should require task-level evidence rather than accepting parameter count or general benchmark scores.

Requests for proposals can ask vendors for watt-hours per successful transaction under a defined workload, carbon-accounting boundaries, data-centre location options, retention and caching policies, and an efficiency improvement plan.

Contracts can establish carbon budgets or intensity targets without encouraging vendors to sacrifice accuracy or safety silently.

8.4. Minimum Green AI Disclosure

Disclosure field	Minimum information
Model	Name, revision, parameter/active-parameter information where available.
Workload	Task, request count, input/output token distribution, context length.
Compute	CPU/GPU/accelerator, quantity, precision, duration, utilisation.
Software	Framework, serving engine, kernels and optimisation settings.
Energy	Measurement source, IT kWh, idle treatment, sampling and uncertainty.
Facility	PUE value/source and included infrastructure.
Carbon	Grid factor, location, time resolution, location- or market-based method.
Quality	Metric, threshold, subgroup tests and failure criteria.
Boundary	Experimentation, training, inference, storage and embodied components included.
Normalisation	Per token, request and successful task plus absolute total.

8.5. Policy Perspective

Policy should encourage comparable disclosure while avoiding rigid assumptions that become obsolete as technology changes. Standards can define measurement boundaries, auditable metadata and uncertainty requirements. Public research funding can reward energy-efficient results, shared infrastructure and open measurement. Because data-centre growth is geographically concentrated, planning must also consider grid capacity, water constraints and community impacts rather than relying only on global electricity shares.

IX. CONCLUSION AND FUTURE RESEARCH

9.1. Conclusion

Green Artificial Intelligence reframes LLM progress around useful capability delivered with the least defensible lifecycle burden. Energy use arises not only in final pre-training but also in exploratory runs, repeated fine-tuning, inference, storage, networking and facility overhead. Carbon footprint depends further on when and where electricity is consumed and on the embodied impact of accelerators. As a result, neither parameter count nor a single energy-per-query estimate can represent sustainability adequately.

The most credible strategy is layered. Researchers must first measure a representative baseline. Developers should then match model capacity to task difficulty, apply compression and efficient adaptation, improve batching and memory management, constrain unnecessary context and output, and choose hardware using measured task-level efficiency. Flexible workloads should be shifted towards lower-carbon electricity where latency, privacy and residency requirements permit. Finally, organisations must monitor absolute demand and publish transparent boundaries so that efficiency improvements are not hidden by rebound growth.

The proposed Green LLM Optimisation Framework converts these principles into six stages—measure, match, minimise, move, monitor and make public. Its central rule is quality-constrained optimisation: an intervention is sustainable only when it reduces lifecycle burden while preserving defined standards of usefulness, safety and equity. This approach avoids both technological pessimism and uncritical scaling. It treats efficiency as a source of scientific rigour, affordability and wider participation in AI research.

9.2. Future Research Directions

- Develop reproducible, open benchmarks that combine task quality, latency, energy, water and lifecycle carbon.
- Measure marginal rather than only average grid emissions for time- and location-aware scheduling.
- Establish validated allocation methods for embodied accelerator emissions under shared cloud use.
- Investigate energy-aware routers that predict the lowest-cost model capable of satisfying each request.
- Study multilingual and low-resource-language quality under quantisation, pruning and distillation.
- Quantify the environmental break-even point for distillation, RAG indexing and hardware replacement.
- Analyse rebound effects by linking per-token efficiency with total organisational demand over time.

- Design privacy-preserving telemetry standards that allow API customers to verify environmental claims.
- Extend assessment from model endpoints to complete agentic workflows, including tool calls, search and repeated self-evaluation.

REFERENCES

[1] Bolón-Canedo, V., Morán-Fernández, L., Cancela, B., & Alonso-Betanzos, A. (2024). A review of green artificial intelligence: Towards a more sustainable future. Neurocomputing, 599, 128096. https://doi.org/10.1016/j.neucom.2024.128096

[2] Dodge, J., Prewitt, T., Tachet des Combes, R., Odmark, E., Schwartz, R., Strubell, E., Luccioni, A. S., Smith, N. A., DeCario, N., & Buchanan, W. (2022). Measuring the carbon intensity of AI in cloud instances. Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency, 1877–1894. https://doi.org/10.1145/3531146.3533234

[3] Faiz, A., Kaneda, S., Wang, R., Osi, R., Sharma, P., Chen, F., & Jiang, L. (2024). LLMCarbon: Modeling the end-to-end carbon footprint of large language models. International Conference on Learning Representations. https://arxiv.org/abs/2309.14393

[4] Henderson, P., Hu, J., Romoff, J., Brunskill, E., Jurafsky, D., & Pineau, J. (2020). Towards the systematic reporting of the energy and carbon footprints of machine learning. Journal of Machine Learning Research, 21(248), 1–43.

[5] International Energy Agency. (2025). Energy and AI. https://www.iea.org/reports/energy-and-ai

[6] Jegham, N., Abdelatti, M., Elmoubarki, L., & Hendawi, A. (2025). How hungry is AI? Benchmarking energy, water, and carbon footprint of LLM inference. arXiv. https://arxiv.org/abs/2505.09598

[7] Lacoste, A., Luccioni, A., Schmidt, V., & Dandres, T. (2019). Quantifying the carbon emissions of machine learning. arXiv. https://arxiv.org/abs/1910.09700

[8] Luccioni, A. S., Viguier, S., & Ligozat, A.-L. (2023). Estimating the carbon footprint of BLOOM, a 176B parameter language model. Journal of Machine Learning Research, 24(253), 1–15.

[9] Masanet, E., Shehabi, A., Lei, N., Smith, S., & Koomey, J. (2020). Recalibrating global data center energy-use estimates. Science, 367(6481), 984–986. https://doi.org/10.1126/science.aba3758

[10] Patterson, D., Gonzalez, J., Le, Q., Liang, C., Munguia, L.-M., Rothchild, D., So, D., Texier, M., & Dean, J. (2021). Carbon emissions and large neural network training. arXiv. https://arxiv.org/abs/2104.10350

[11] Patterson, D., Gonzalez, J., Hölzle, U., Le, Q., Liang, C., Munguia, L.-M., Rothchild, D., So, D., Texier, M., & Dean, J. (2022). The carbon footprint of machine learning training will plateau, then shrink. Computer, 55(7), 18–28. https://doi.org/10.1109/MC.2022.3148714

[12] Schwartz, R., Dodge, J., Smith, N. A., & Etzioni, O. (2020). Green AI. Communications of the ACM, 63(12), 54–63. https://doi.org/10.1145/3381831

[13] Strubell, E., Ganesh, A., & McCallum, A. (2019). Energy and policy considerations for deep learning in NLP. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 3645–3650. https://doi.org/10.18653/v1/P19-1355

[14] Wu, C.-J., Raghavendra, R., Gupta, U., Acun, B., Ardalani, N., Maeng, K., Chang, G., Behram, F. A., Huang, J., Bai, C., Gschwind, M., Gupta, A., Ott, M., Melnikov, A., Candido, S., Brooks, D., Chauhan, G., Lee, B., Lee, H.-H. S., ... Hazelwood, K. (2022). Sustainable AI: Environmental implications, challenges and opportunities. Proceedings of Machine Learning and Systems, 4, 795–813.

[15] You, J., Chung, Y., & Chowdhury, M. (2023). Zeus: Understanding and optimizing GPU energy consumption of DNN training. 20th USENIX Symposium on Networked Systems Design and Implementation, 119–139.