What is the environmental impact of large language models (LLMs), and how can these same models be leveraged to reduce global carbon footprints?

Introduction

Climate change is among the most pressing challenges facing the global community today, requiring immediate and coordinated responses across all sectors of society. Digital technology, notably artificial intelligence (AI), is simultaneously recognized as both a significant contributor to greenhouse gas emissions and as a powerful tool for addressing environmental challenges. Within the field of AI, Large Language Models (LLMs)—advanced computational models trained on vast datasets to understand and generate human-like text—have risen dramatically in prominence and application, influencing countless aspects of modern life, from digital communication to industrial optimization.

The environmental impact of these models, however, remains poorly understood by the broader public and even within many sectors of the technology industry. Their training and operation demand immense computational resources, translating into considerable energy consumption and, consequently, substantial carbon emissions. Yet paradoxically, these same technologies hold potential for significant positive environmental impacts, capable of optimizing energy use, enhancing sustainability initiatives, and supporting climate change mitigation efforts.

This thesis aims to critically analyze the dual role of Large Language Models in the context of climate change. Specifically, it addresses the central question: *"What is the environmental impact of large language models, and how can these same models be leveraged to reduce global carbon footprints?"* By systematically examining both the negative impacts associated with their development and operation, as well as their capacity for positive environmental contributions, this work seeks to offer a balanced and insightful perspective on the role of LLMs in contemporary environmental challenges. Ultimately, this exploration aims to inform sustainable practices within the field of AI, highlighting pathways for developers, policymakers, and corporations to responsibly harness the power of large-scale computational models for a more sustainable future.

2.1 Large Language Models (LLMs)

As briefly introduced, Large Language Models (LLMs) leverage advanced architectures to provide unprecedented language capabilities. This section elaborates comprehensively on their structures, methodologies, and practical uses

2.1.1 Definition and Core Architecture:

Definition and Scale. Large language models (LLMs) are neural network models with hundreds of millions to tens of billions of tunable parameters. They are trained on vast corpora of text to capture the statistical structure and semantics of human language. Modern LLMs learn in two main stages. In the first stage—self-supervised pre-training—the model is trained on unlabeled text to predict parts of the input from other

parts. Common pre-training objectives include autoregressive language modeling, where the model predicts the next token given the previous context, prefix language modeling, where a randomly chosen prefix is used and the rest of the tokens are predicted, and masked language modeling, where random tokens or spans are masked and the model learns to reconstruct them. Pre-training teaches the model general linguistic patterns and world knowledge. In the second stage—fine-tuning—the model is adapted to a specific task (e.g., question answering, summarization or instruction-following) using labeled data. Fine-tuning may take the form of task-specific transfer learning, instruction tuning, or alignment with human feedback. This two-stage training paradigm allows a pre-trained model to be reused for many downstream tasks with relatively little additional data.

The core architectural innovation enabling modern LLMs is the **Transformer**, introduced in 2017. Unlike earlier recurrent neural networks (RNNs) or long short-term memory (LSTM) networks, Transformers rely entirely on self-attention to model relationships among tokens in a sequence. Self-attention assigns a relevance score to each token in a sequence based on the current context, allowing the model to focus on the most informative words. For example, in the sentence “Swing the bat! … The bat flew at night,” the attention mechanism weights the word “bat” differently in each context, enabling the model to distinguish between the sports equipment and the animal. Self-attention is formulated through query, key and value vectors: each token emits a query, key and value; attention scores are computed by taking the scaled dot product of queries with keys; and the resulting weights are used to aggregate values. This design captures both local and long-range dependencies and allows the model to consider global context simultaneously. Moreover, Transformers use multi-head attention, which runs several self-attention operations in parallel with different learned projections. Multi-head attention allows the model to attend to different aspects of the context at once, enhancing robustness and expressivity. Because there are no recurrent connections, Transformers can process all tokens in a sequence in parallel, greatly accelerating training and enabling models to scale to billions of parameters .

The elimination of recurrence and reliance on self-attention makes Transformers highly scalable. Each input token can attend to every other token in a layer, but these attention computations can be parallelized across all positions. This parallelism allows Transformer-based LLMs to train efficiently on large datasets and to model long-range context more effectively than RNN-based architectures. It also enables the use of distributed training across many GPUs or TPUs. Empirical scaling laws show that as the model size, dataset size and compute budget grow, the model’s performance improves according to a power law, motivating the trend towards ever-larger LLMs. However, the attention mechanism has a quadratic complexity with respect to sequence length, so training and inference cost grows rapidly with longer contexts. This tension between expressivity and computational cost is a central theme in the evolution of LLMs and sets the stage for the historical overview that follows.

2.1.2 Evolution and Key Milestones

The journey to today’s large language models spans decades of research. Early language models relied on statistical n-gram models, which estimated the probability of the next word based on a fixed-length context. Although simple, n-gram models could not capture long-range dependencies or complex syntax. In the 1980s and 1990s, recurrent neural networks (RNNs) and their more robust cousin, the long short-term memory (LSTM) network, improved sequence modeling by maintaining a hidden state across time steps. LSTMs addressed the vanishing gradient problem and performed well for moderate-length sequences, yet they still struggled to capture dependencies over hundreds of tokens and were difficult to parallelize.

A pivotal moment came in 2017 with the introduction of the Transformer architecture by Vaswani et al. in the paper **“Attention Is All You Need”**. The Transformer overcame many limitations of RNN/LSTM models by relying entirely on self-attention and positional encoding, enabling non-sequential, parallel processing. This innovation allowed researchers to train much larger models on massive datasets and to model long-range dependencies effectively. The combination of self-attention’s efficiency and the scaling properties of neural networks led to an explosion of model sizes and capabilities.

Following the Transformer’s debut, a sequence of pre-trained LLMs marked key milestones. In 2018, Google introduced BERT (Bidirectional Encoder Representations from Transformers), a model that used masked language modeling and next-sentence prediction during pre-training to learn deep bidirectional representations of text. BERT could be fine-tuned with just one additional output layer to achieve state-of-the-art performance across a range of tasks. In the same era, OpenAI released the Generative Pre-Trained Transformer (GPT) series, which used an autoregressive pre-training objective to generate text. GPT-2 (2019) expanded to 1.5 billion parameters and exhibited striking fluency, while GPT-3 (2020) scaled to 175 billion parameters and demonstrated emergent abilities such as few-shot learning, where the model performs a new task from only a handful of examples—capabilities not apparent in smaller models. These releases showcased how increasing parameter counts and dataset sizes could lead to qualitatively new behaviours, confirming the scaling laws.

The subsequent years saw a proliferation of LLMs from both industry and academia, each pushing the boundaries of size and efficiency. Google’s T5 recast all NLP tasks into a text-to-text format; XLNet introduced permutation-based training to capture better bidirectional context; RoBERTa showed that simply training BERT longer and on more data yields improvements; and open-source projects like GPT-Neo and Meta’s LLaMA democratized access to powerful models. By late 2022, conversational agents like ChatGPT, built on GPT-3.5 and later GPT-4, brought LLM capabilities to the mainstream, prompting a surge of applications across sectors.

This rapid technical evolution illustrates how architectural innovations (Transformers) and scaling strategies (massive pre-training followed by task-specific fine-tuning) have transformed natural language processing.

2.1.3 Capabilities and Applications:

Having traced how advances in architecture and scale gave rise to modern LLMs, we now turn to what these models can actually do. A defining feature of LLMs is their general-purpose language ability. Because they are pre-trained on diverse, Internet-scale text corpora, LLMs acquire a broad knowledge base that enables them to perform a wide variety of tasks. At the most fundamental level, a single LLM can:

- Generate coherent text, from essays and articles to creative stories and poetry;
- Summarize long documents or synthesize multiple sources into concise overviews;
- Translate between languages and dialects;
- Answer questions or engage in multi-turn dialogues;
- Reason over information given in textual form to solve problems or follow chains of logic.

In contrast to earlier specialized models, these diverse abilities come from a single model that can be adapted through fine-tuning or prompting. Researchers therefore refer to them as “foundation models”: they form a general base that can be reused across many downstream applications. Moreover, as the previous section noted, increasing model size and data has led to emergent capabilities: abilities that seem to appear suddenly once models reach a certain scale. For example, recent work observes that some tasks (such as three-digit addition) suddenly become solvable only after the model is scaled past a threshold, a

phenomenon sometimes described as “emergence”. Although the existence and interpretation of such jumps is debated, the key point is that very large LLMs can perform in-context learning—that is, they can follow instructions or learn from a handful of examples provided in the prompt without additional training—a behavior not observed in smaller models. This versatility makes LLMs uniquely powerful tools.

The general-purpose capabilities of LLMs translate into concrete benefits across industries. A recent review in Scientific Reports highlights the following examples:

- **Healthcare**: LLMs assist in diagnosing diseases, personalizing treatment plans and managing patient data. They can summarize clinical notes and literature to support diagnostic decision-making and help identify adverse events.
- **Automotive**: LLMs support predictive maintenance by analyzing maintenance logs and sensor data; they also power in-car virtual assistants that understand spoken commands and provide real-time translation, improving safety and user experience.
- **E-commerce and Consumer Services**: LLMs drive recommendation systems, analyze consumer behavior and power chatbots that handle customer queries, optimize search results and personalize shopping experiences.
- **Education**: LLMs facilitate personalized learning through intelligent tutoring systems, automated grading and tailored feedback, making education more accessible and effective.
- **Finance and Banking**: LLMs are used for fraud detection, customer-service automation and risk management. They can parse financial news, generate reports and identify anomalies in transaction data.

Across these domains, LLMs *“drive significant advancements by automating tasks, improving accuracy and providing deeper insights”*. Put simply, organizations deploy LLMs to streamline workflows, enhance customer interactions and unlock data-driven insights—often matching or surpassing human performance on specialized language tasks.

Taken together, these capabilities and use cases show why the evolution described previously matters. The architectural innovations that enabled LLMs have not merely produced larger models; they have produced tools with unparalleled versatility and real-world impact.

2.1.3 Toward Environmental Impact:

While LLMs enable remarkable capabilities, they also introduce significant sustainability challenges. Performance gains from scaling model size come with steep compute and energy costs. A landmark 2019 analysis quantified how training modern NLP models can emit **hundreds of thousands of pounds of CO_2**; in its most extreme case, the team estimated **≈626,000 lb (≈284 tCO_2e)** for a single training pipeline with extensive hyperparameter search, bringing the environmental cost of “bigger is better” into sharp relief. More recently, widely cited estimates for GPT-3 (175B) put its training energy around **1,287 MWh** and **≈552 tCO_2e**, underscoring the scale of resources required for state-of-the-art models. Larger successors (e.g., GPT-4) are believed to have required substantially more compute, though precise figures remain undisclosed; emerging work also highlights non-carbon impacts such as significant cooling-water use during training.

Operational use compounds these concerns. Once deployed at scale, LLMs answer millions to billions of queries, and inference—the forward pass to serve user requests—often dominates lifetime emissions. Analyses across providers and academic studies now converge on this point: inference typically outweighs training over a model’s service life, with the balance shaped by prompt length, tokens generated,

model/hardware choice, and data-center carbon intensity. Recent modeling suggests that typical ChatGPT-class requests consume on the order of ~0.3 Wh per query (far lower than earlier, rougher heuristics), though the true value varies with usage patterns and systems. The key takeaway is that aggregate demand —rather than one-off training—drives much of the ongoing footprint.

These dynamics frame the central question for the rest of this thesis: **how do we measure, compare, and ultimately reduce** LLM emissions across training and inference without forfeiting their benefits? We turn next to the methods and standards used to quantify energy use and translate it into CO_2-equivalent—laying the groundwork for evaluating models, infrastructure choices, and mitigation strategies in a consistent way.

Section 2.2: Current Methods and Standards for Measuring the Environmental Impact of Computing

In this section, we review how the environmental footprint of computing is assessed, focusing on energy use and greenhouse gas (GHG) emissions. As Large Language Models (LLMs) demand massive computation, understanding these measurement methods is crucial to quantify their environmental impact. We cover how energy consumption is measured, how it is translated into CO_2-equivalent emissions, the tools and frameworks available, and the relevant standards (e.g. GHG Protocol, ISO 14064) for reporting such impacts.

2.2.1 Measuring Energy Consumption in Computing

Assessing the climate impact of AI begins with measuring how much electricity computing workloads consume. At the smallest scale, researchers use on-board sensors or power APIs to sample the real-time power draw of CPUs and GPUs. Utilities such as NVIDIA's nvidia-smi and Intel's RAPL interface allow developers to read power draw every few milliseconds and integrate it over time to obtain total energy in kilowatt-hours (kWh). *Strubell et al. (2019)*, for example, instrumented the GPUs and CPUs used in NLP experiments and multiplied the average power by runtime to estimate training energy. This direct measurement approach can be extended to memory, storage and networking equipment; summing the energy of all components (including idle and overhead energy) yields the total electricity consumption of a computing task.

Data-Center Overhead and PUE

In large-scale computing facilities, not all of the electricity drawn from the grid goes into running chips; cooling systems, power conversion, lighting and other infrastructure add overhead. The industry standard metric for capturing this overhead is Power Usage Effectiveness (PUE)—the ratio of total facility energy to the energy consumed by IT equipment. A PUE of 1.58 (the approximate global average around 2018) means that for every 1 kWh delivered to servers, another 0.58 kWh is consumed by cooling and other supporting systems. Companies publicly report their PUE to demonstrate efficiency improvements. For example, Google reports a fleet-wide PUE of 1.09 across its large-scale data centres, meaning only 9% overhead; this is significantly lower than the industry average of around 1.56. Google's quarterly PUE data show values between 1.04 and 1.14 across individual campuses. Using PUE in emissions calculations simply multiplies the measured IT energy by the PUE factor to account for total facility electricity use.

Life-Cycle Considerations

Operational energy is the most visible contributor to a data centre's footprint, but a comprehensive assessment must include embodied emissions—the energy and materials used to manufacture and construct data-centre hardware and buildings. A 2024 report summarised by TechRadar, citing Morgan Stanley, estimated that about 60 % of future data-centre emissions will come from operations and roughly 40 % from the construction of facilities and infrastructure. A life-cycle assessment (LCA) therefore examines *"cradle to grave"* impacts: raw material extraction, chip fabrication, facility construction, operational energy, maintenance, and end-of-life decommissioning. Data4 Group's LCA of European data centres illustrates this approach: they found that a typical facility produces 6,600–10,400 tonnes of CO_2 per megawatt of operational IT over a 20-year period, and that 80 % of those emissions come from energy used in operations, while the remaining emissions arise from construction and materials. Construction emissions alone amount to 1,500–2,100 tCO_2 per megawatt, and within that footprint the concrete and steel used in the building shell contribute about 25 %. These LCAs show why focusing solely on operational energy can underestimate the true environmental cost of computing.

Because manufacturing data for chips and servers are often proprietary, practitioners sometimes rely on typical power ratings or thermal design power (TDP) and assume average utilisation to approximate embodied energy. Combining such estimates with the Software Carbon Intensity (SCI) framework—which defines emissions per unit of software function—allows researchers to compare models or services on a per-query basis. Overall, measurement frameworks that pair direct power monitoring (for training and inference) with facility-level metrics (PUE) and cradle-to-grave LCAs provide the foundations for rigorous, transparent assessments of AI's environmental impact.

2.2.2 Converting Energy Use to CO_2-Equivalent Emissions

Emission Factors and CO_2e Calculation

Once the total electricity consumption of a training run or inference workload is known, it must be converted into carbon dioxide equivalent (CO_2e) to quantify climate impacts. This conversion multiplies energy (kWh) by an appropriate emissions factor (kg CO_2e per kWh), which represents the carbon intensity of the electricity supply. Emissions factors vary by region: a grid dominated by coal has a high factor, while one supplied largely by wind or hydro has a low factor. CodeCarbon, for example, defaults to a global average of 475 g CO_2e/kWh (0.475 kg CO_2e/kWh) when country-specific data are unavailable. Researchers often select factors using regional electricity mixes or cloud providers' disclosures; if the energy mix is known (e.g., 50 % renewable and 50 % gas), a weighted average factor is applied. The basic conversion is therefore:

$$\mathrm{COe(kg)} = \mathrm{Energy(kWh)} \times \mathrm{Emissionfactor(kg/kWh)}.$$

When accounting for data-centre overhead and multiple devices, a more detailed formula is used. Stanford's CS324 course expresses total emissions as the product of three terms: the energy-to-emissions conversion factor `Rpower -> emit`, the power usage effectiveness (PUE) of the data centre, and the sum of each device's power multiplied by its runtime. Mathematically:

$$\mathrm{COe} = R_{\text{power}\to\text{emit}} \times \mathrm{PUE} \times \sum_{\text{devices}} P_{\text{device}} \times t.$$

This equation ensures that one accounts for not only the IT load but also the extra electricity used for cooling and power distribution (via PUE). Using a region-appropriate emission factor is critical; the same model trained on a renewable-powered grid will emit much less CO_2e than one trained on a fossil-heavy grid.

Model	Transformer (Big)		Evolved Transformer (Medium)	Transformer (Big)	Evolved Transformer (Medium)
Number of Parameters (B)	0.21		0.13	0.21	0.13
Datacenter	US Average	Google Iowa Council Bluffs			
Datacenter Gross CO_2e/KWh (kg/KWh) 2020 (Section 2.4 and Appendix D)	0.429	0.478			
Datacenter Net CO_2e/KWh (kg/KWh) 2020 (Section 2.4 and Appendix D)	0.429	0.080			
Datacenter PUE (Latest quarter 2020)	1.59	1.11			
Processor	P100			TPU v2	
Chip Thermal Design Power (TDP in Watts)	300			280	
Measured System Average Power including memory, network interface, fans, host CPU (Watts)	296		271	229	227
Measured Performance (TFLOPS/s)[5]	6.7		4.7	28.8	24.0
Number of Chips	8				
Training time to accuracy goal (days)	3.5		3.2	0.81	0.62
Total Computation (floating point operations)	1.61E+19		1.03E+19	1.61E+19	1.03E+19
Energy consumption (KWh)	316	221	185	40	30
Gross CO_2e for Model Training (metric ton) (Section 2.4 and Appendix D)	0.1357	0.1055	0.0883	0.0189	0.0143
Net CO_2e for Model Training (metric ton) (Section 2.4 and Appendix D)	0.1357	0.0177	0.0148	0.0032	0.0024
% 24/7 net carbon free energy (CY 2019)	N/A	78%			

Google's Emissions Factors Table, showing regional and energy source-specific factors.

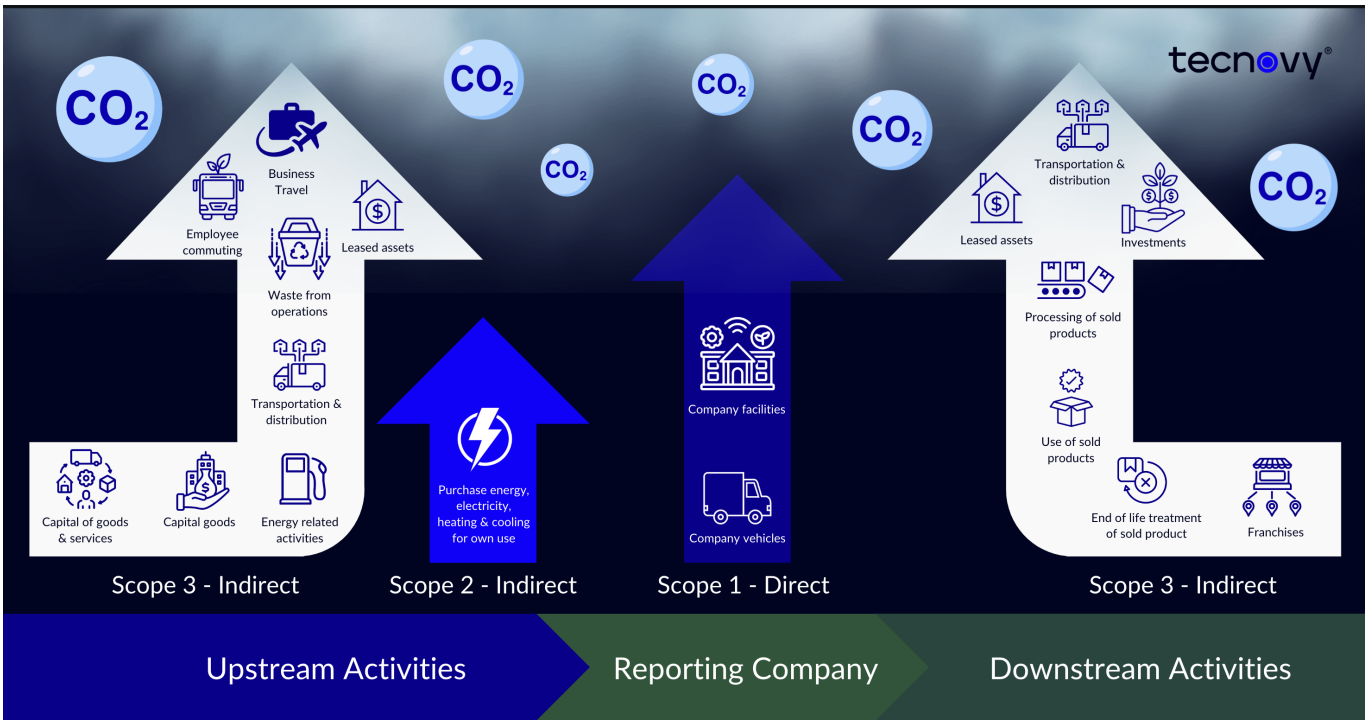
Origins and adoption of the GHG Protocol

To contextualize these calculations, it is useful to recall that the **Greenhouse Gas (GHG) Protocol** provides the overarching framework used by most organizations to measure and report emissions. The protocol was launched in 1998 as a collaboration between the **World Resources Institute (WRI)** and the **World Business Council for Sustainable Development (WBCSD)** to develop internationally accepted GHG accounting standards. Its first Corporate Standard appeared in 2001 (updated in 2004) and introduced the widely used Scope 1, Scope 2 and Scope 3 categories that now underpin corporate climate disclosures. Adoption has become near-universal: the GHG Protocol Secretariat reports that 97 % of S&P 500 companies disclosing to CDP in 2023 used its methods, and CDP received climate disclosures from more than 18 700 companies in 2022, a sharp increase over previous years.

Legal frameworks referencing the GHG Protocol.

Although the protocol itself is not statutory, it is embedded in several regulations. The European **Corporate Sustainability Reporting Directive (CSRD)** and its **European Sustainability Reporting Standards (ESRS)** require large firms to report gross Scope 1, Scope 2 and Scope 3 emissions, explicitly referencing the GHG Protocol's Corporate and Scope 3 standards and even requiring companies to include emissions

from purchased cloud computing and data-centre services in upstream Scope 3 when material. The global **IFRS S2** climate-disclosure standard likewise mandates Scope 1–3 measurement in accordance with the GHG Protocol and is expected to cover **100 000–130 000 companies** across multiple jurisdictions. At the national level, California's Climate Corporate Data Accountability Act will require companies with revenues over USD 1 billion to report Scope 1 and Scope 2 emissions from **2026** and Scope 3 from **2027**, effectively pushing large U.S. firms toward GHG-compliant accounting. These developments highlight how the GHG Protocol has become the de facto framework for regulated climate disclosures, providing common definitions and guidance for computing emissions across all scopes.



Schema of the GHG Protocol, showing Scope 1, 2 and 3 emissions.

Beyond Carbon

While CO_2 remains the primary focus of climate accounting, other environmental metrics are increasingly recognized. One such metric is water usage, because data centres rely on large volumes of water for cooling and, in some regions, for electricity generation. The industry standard for measuring cooling-water efficiency is **Water Usage Effectiveness (WUE)**, defined as litres of water consumed per kilowatt-hour (kWh) of IT energy. Average data centres have a WUE around **1.8 L kWh^{-1}**, whereas Amazon Web Services claims a much lower **0.15 L kWh^{-1}**, and Microsoft reports an average **0.30 L kWh^{-1}**. WUE varies by climate; for example, Microsoft's portfolio spans values from **1.52 L kWh^{-1}** in arid Arizona to **0.02 L kWh^{-1}** in humid Singapore. As AI workloads proliferate, water demand is rising: training a single large language model in a Microsoft data centre can evaporate about **700 000 litres** of water, and global AI demand could require **4.2–6.6 billion m³** of water withdrawal by 2027. These findings have prompted sustainability frameworks to encourage reporting of water use alongside CO_2 emissions and to promote water-saving measures—such as closed-loop cooling, situating facilities in cooler climates, and using reclaimed or recycled water.

A second emerging impact metric is electronic waste (e-waste). LLMs and generative AI demand constant hardware upgrades, leading to high turnover of servers, printed circuit boards and batteries that can contain toxic substances like lead and chromium. A recent Physics World analysis estimated that, without mitigation, generative AI could produce **2.5 million tons** of e-waste annually by 2030, and in a worst-case scenario the total e-waste generated between 2023 and 2030 could reach **5 million tons**. Rapid server

turnover and geopolitical pressures on semiconductor supply chains would exacerbate this trend. However, strategies such as extending the lifespan of computing infrastructure, reusing or remanufacturing components, and improving recycling could reduce e-waste by **up to 86 %**. These results illustrate that carbon-only metrics do not capture the full environmental footprint of AI infrastructure. Consequently, sustainability assessments are increasingly calling for water, e-waste and other impacts to be reported alongside CO_2e.

2.2.3 Tools and Frameworks for Carbon Accounting in Computing

Accurately recording the energy consumed by computing workloads and converting those numbers to greenhouse-gas emissions requires a combination of tools and methods. Recent work has expanded this ecosystem to cover everything from embedded sensors on CPUs to cloud-wide dashboards, and many initiatives now consider operational emissions and embodied hardware emissions separately.

Software Instrumentation Libraries

To understand the energy use of a computing workload one must monitor the hardware itself. Low-level instrumentation libraries have emerged that integrate with machine-learning code to sample power counters and convert the resulting measurements into estimates of CO_2-equivalent (CO_2e) emissions. CodeCarbon is widely adopted: it attaches to the CPU, GPU and RAM, multiplies their consumption by location-specific emission factors and displays the resulting CO_2e in a dedicated dashboard that even recommends cloud regions with a lower carbon intensity. CarbonTracker takes a similar approach but emphasises real-time monitoring; it queries a carbon-intensity API based on the user's geographic location, supports multiple chip architectures and updates its predicted emissions after a few training epochs with minimal overhead.

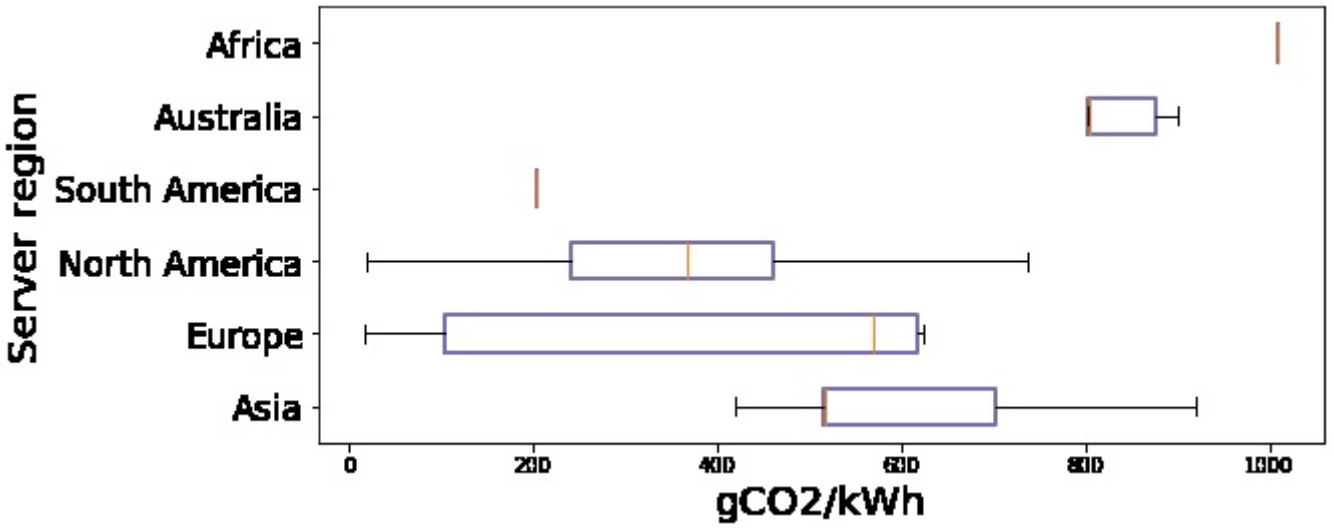
Ease of integration is a recurring theme. Eco2AI wraps the measurement and logging functionality in a simple decorator that can be applied to existing functions; the library records energy consumption per run, multiplies it by a regional emission coefficient and falls back to a global average when the country is unknown. At the other end of the spectrum, pyJoules exposes the lowest-level counters available: it uses the Intel Running Average Power Limit (RAPL) interface and Nvidia Management Library (NVML) to measure energy drawn by CPUs, memory and GPUs. When users decorate a function, pyJoules reports its energy consumption directly and cautions that extraneous background tasks should be disabled to obtain accurate measurements.

Because each library must access hardware power sensors, they remain confined to bare-metal systems or privileged virtual machines. They cannot yet account for the embodied emissions of manufacturing the hardware itself or measure workloads running on managed cloud services, where only aggregated telemetry is available.

Carbon Calculators

When direct measurement is impractical—because the computation runs on a shared server, a managed cloud environment or an HPC scheduler—researchers turn to carbon calculators. These tools trade precision for accessibility by approximating emissions from high-level inputs. The ML CO_2 Impact Calculator asks users to specify the type and number of GPUs used and the duration of the run; it then estimates the resulting CO_2 emissions and indicates how much of that figure has already been offset by the chosen cloud provider. Since the tool does not account for the power-usage effectiveness (PUE) of the data centre, users are encouraged to adjust the result by the PUE of their facility. Its emissions factors come from

electricityMap and other published datasets, and the estimates are therefore based on annual average grid mixes.



Variation of the Average Carbon Intensity of Servers Worldwide, by Region. (Vertical bars represent regions with a single available data point.)

The Green Algorithms model generalises this idea beyond GPUs. By combining three inputs—runtime, hardware specification (CPU, GPU and memory) and country of execution—the calculator estimates energy consumption and converts it to CO_2e emissions. To help users grasp the magnitude of their impact, the output also expresses the result in "tree-months," a heuristic indicating how long it would take a tree to absorb the same amount of carbon. The Green Algorithms methodology is open and has been extended into Green Algorithms 4 HPC, which automatically scans job records on high-performance clusters and provides aggregate estimates.

Because carbon calculators rely on simplified models and average utilisation assumptions, their results are best used as approximate indicators or to raise awareness in research papers rather than as detailed accounting. They cannot capture dynamic power scaling, variations across datacentres or the embodied emissions of hardware manufacture.

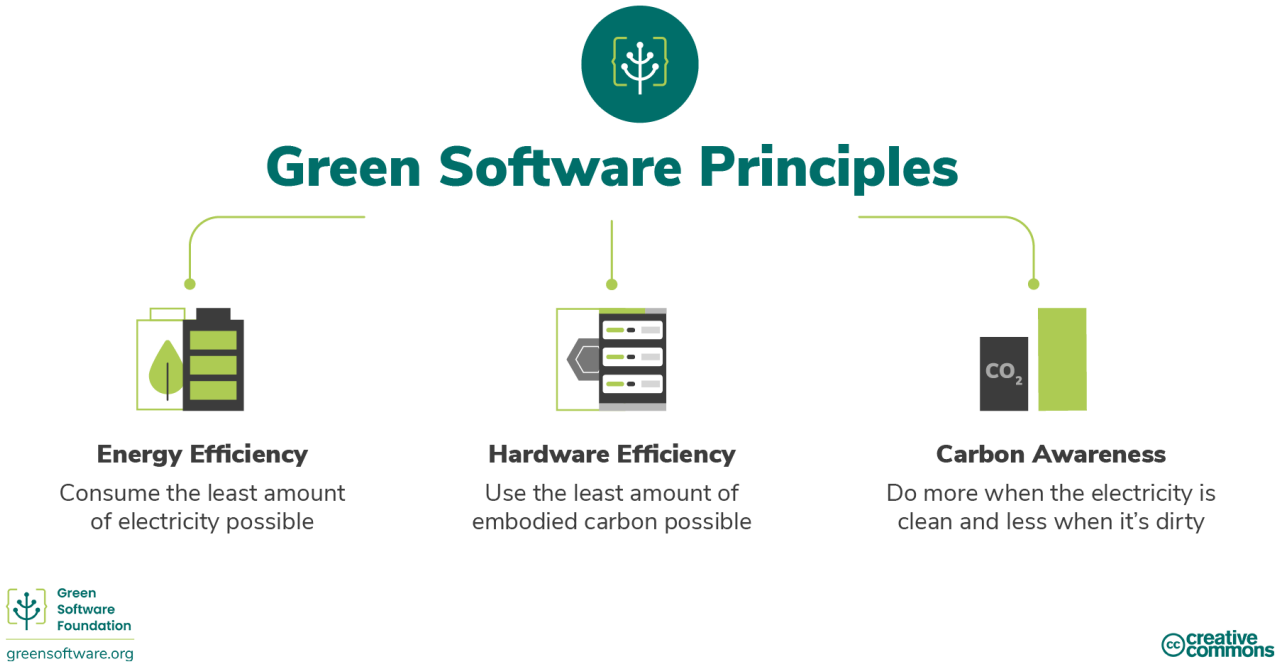
Cloud Provider Sustainability Tools

Major cloud providers have introduced dashboards for clients to track the footprint of their cloud usage. For example, Microsoft Azure's Sustainability Calculator (a Power BI application) helps organizations measure the carbon impact of their Azure cloud services. It aggregates a company's cloud resource usage and applies Microsoft's data on data center PUE and energy sourcing to estimate emissions. These tools leverage internal telemetry and "industry-standard emission factors" to give relatively accurate reports. Google and Amazon have similarly provided carbon footprint tools for their cloud platforms. Such tools are particularly useful for companies to track Scope 2 emissions of their cloud-based AI workloads as part of corporate GHG reporting (aligning with GHG Protocol and sustainability goals).

Experiment Tracking and Reporting Frameworks

Within the AI research community, frameworks have emerged to encourage reporting of energy and emissions. For instance, Experiment Impact Tracker (Peter Henderson) is a toolkit that logs energy use of ML training runs and produces a report of carbon impact, integrating with training scripts. Some academic

conferences now suggest or require submitting energy usage alongside model performance. The Green Software Foundation has even proposed a Software Carbon Intensity (SCI) score, which quantifies emissions per unit of software function.



Green Software Foundation's principles.

2.2.4 Standards and Protocols for Environmental Impact Measurement

The field of computing sustainability borrows from general environmental accounting standards, ensuring that measurements are credible and comparable:

ISO 14064 Series ISO 14064-1:2018 is an international standard that specifies principles and requirements for quantifying and reporting an organization's GHG emissions and removals. It aligns closely with GHG Protocol and adds a layer of formal verification. Academic and industry studies on computing impact sometimes reference ISO 14064 to demonstrate rigor in carbon accounting. For instance, a white paper might claim compliance with ISO 14064 to ensure the carbon footprint of an AI service was quantified following accepted principles (e.g. completeness, transparency). The ISO 14064-2 standard focuses on project-level emissions reductions, and could be relevant if a new AI model or data center design is claimed to reduce emissions, it guides how to calculate that reduction. ISO 14064-3 covers third-party validation of emissions reports, which becomes important as companies publishing AI models may seek independent assurance of their environmental claims. In summary, ISO 14064 provides a structured, internationally recognized approach to measuring GHG emissions, and is the go-to standard for ensuring consistency and credibility in emission numbers.

ISO 14040/14044 (Life Cycle Assessment Standards) These ISO standards outline how to conduct a Life Cycle Assessment, which, as noted above, is crucial for evaluating environmental impacts beyond just electricity use. (*"Unraveling the Hidden Environmental Impacts of AI Solutions for Environment Life Cycle Assessment of AI Solutions"* by Anne-Laure Ligozat, Julien Lefèvre, Aurélie Bugeau and Jacques Combaz) emphasize using LCA to capture full impacts of ML models. The standards guide practitioners to define system boundaries, inventory all relevant emissions (from manufacturing to disposal), and assess multiple

impact categories. In applying ISO 14040 to computing, one must consider the lifecycle of IT equipment and infrastructure supporting AI. While comprehensive, performing a full LCA for each new LLM is often impractical, but the standard serves as an aspirational framework to identify otherwise "hidden" impacts (like hardware production or coolant chemicals) that purely operational metrics might miss.

Emerging Software-Specific Standards Recognizing the need for IT-specific guidance, new standards are being developed. A notable example is ISO/IEC 21031:2024, a fresh standard focusing on the carbon footprint of software systems. It introduces the concept of a Software Carbon Intensity (SCI) score as a standardized measure of how much CO_2 is emitted per unit of software function. This standard essentially formalizes the methodology for calculating emissions attributable to software, covering steps like defining the software boundary (what parts of the software/hardware stack to include), measuring energy consumption, and determining the carbon intensity of the energy source. The SCI framework, developed in part by the Green Software Foundation, aligns with GHG Protocol/ISO 14064 principles but tailors them to software and cloud services. As AI applications (like LLM-driven services) are essentially software delivered at scale, ISO 21031 could soon provide a consistent way to report the carbon footprint per 1000 queries of an LLM service, for example. This represents a move toward standardizing "green software" metrics to complement traditional hardware-focused metrics.

Other Relevant Metrics and Certifications In practice, several industry metrics help evaluate computing efficiency which indirectly relate to environmental impact. For example, PUE (discussed above) is promoted by The Green Grid consortium and is widely adopted by data center operators as a key environmental efficiency metric. There are also certifications like ENERGY STAR, and data center sustainability certifications (LEED, ISO 50001 for energy management) which, while not specific to AI, ensure that infrastructure meets certain energy efficiency criteria. These standards and certifications create a context in which the environmental impact of LLMs should be interpreted – e.g., an AI lab running models in an ENERGY STAR-certified data center with a low PUE and 100% renewable energy procurement will have a much smaller GHG footprint than one using older, inefficient facilities.

2.2.5 Application of These Methods and Standards to AI and LLMs

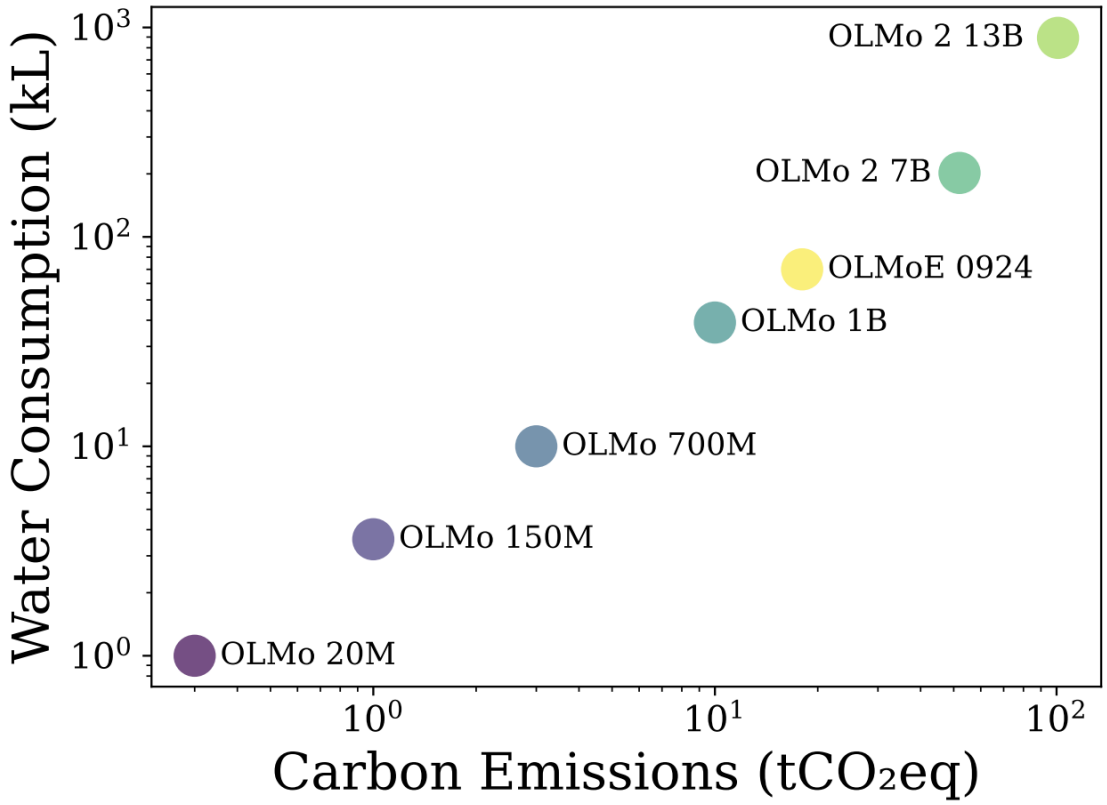
Having outlined general methods, we now connect them to AI/LLM-specific scenarios. The energy and carbon footprint of Large Language Models are typically assessed at two key stages: **training and inference (deployment)**.

Training Phase Footprint Training a state-of-the-art LLM involves running tens of thousands of GPU hours, making energy usage enormous. Researchers apply the measurement techniques above to quantify this. For example, a landmark study by Emma Strubell, Ananya Ganesh, Andrew McCallum. (2019) measured the energy consumed in training several NLP models and estimated their CO_2 emissions.

Model	Hardware	Power (W)	Hours	kWh·PUE	CO_2e	Cloud compute cost
$\text{Transformer}_{base}$	P100x8	1415.78	12	27	26	$41–$140
Transformer_{big}	P100x8	1515.43	84	201	192	$289–$981
ELMo	P100x3	517.66	336	275	262	$433–$1472
BERT_{base}	V100x64	12,041.51	79	1507	1438	$3751–$12,571
BERT_{base}	TPUv2x16	—	96	—	—	$2074–$6912
NAS	P100x8	1515.43	274,120	656,347	626,155	$942,973–$3,201,722
NAS	TPUv2x1	—	32,623	—	—	$44,055–$146,848
GPT-2	TPUv3x32	—	168	—	—	$12,902–$43,008

Estimated cost of training a model in terms of CO2 emissions (lbs) and cloud compute cost (USD).7 Power and carbon footprint are omitted for TPUs due to lack of public information on power draw for this hardware.

They found that training a large transformer with hyperparameter tuning emitted on the order of hundreds of thousands of pounds of CO_2 (over 280 metric tons), roughly equivalent to five cars' lifetime emissions. The procedure included monitoring GPU power draw, summing total kWh, and multiplying by a carbon factor. More recent LLM training reports (for models like GPT-3, Llama, etc.) similarly calculate emissions by recording total energy used. Notably, Meta's Llama model reports listed the electricity consumed and then converted to emissions using region-specific factors (though early versions used rough averages). In a comprehensive 2025 study, Jacob Morrison, Clara Na, Jared Fernandez, Tim Dettmers, Emma Strubell, Jesse Dodge. measured the training of a series of language models and explicitly followed GHG Protocol Scope 2 methods, determining that training their 13-billion-parameter model consumed hundreds of MWh and emitted 493 metric tons of CO_2e (assuming typical U.S. grid mix). These examples illustrate how standard energy-to-emission conversion is applied to quantify the climate impact of training LLMs.



(from ["Holistically Evaluating the Environmental Impact of Creating Language Models."])Models are ordered by their total water consumption and associated CO_2 emissions. Sub-billion-parameter systems were trained on 1.7 trillion tokens; OLMo 1B on 3 trillion; OLMo 2-7B on 4 trillion; OLMoE on 5 trillion; and OLMo 2-13B on 5.6 trillion. The data show that environmental impact rises sharply as both model size and training-data volume increase.

Inference and Deployment Footprint Once deployed, LLMs can be used millions or billions of times, so the per-query energy becomes critical. Methods for measuring inference energy mirror those for training: instrument the model serving hardware to measure power per query, or estimate via benchmarked power usage. Some works use a functional unit like “per 1000 queries” to report emissions, aligning with the SCI approach of emissions per operation. For instance, an online inference energy tool by Hugging Face reports how many watt-hours each API call uses. By multiplying that by the carbon intensity of the host server’s electricity, one obtains CO_2e per query. A study by Alexandra Sasha Luccioni, Sylvain Viguier, Anne-Laure Ligozat. *"Estimating the carbon footprint of bloom, a 176B parameter language model"* evaluated the footprint of serving NLP models and emphasized considering the potentially vast number of inferences, which can quickly match or exceed training emissions. Indeed, if an LLM is very popular, the cumulative electricity for inference (across all user queries) can rival the training cost within months. This highlights the need to measure environmental impact across the model lifecycle, training (one-time, but intensive) and inference (continuous). Standards like the GHG Protocol would count both under the service provider’s Scope 2 emissions, and recent research encourages reporting both stages.

Using Standards in Practice The methodologies and standards described are not just theoretical, they are increasingly being adopted by AI practitioners. For example, when OpenAI or Google report on their models’ sustainability, they often cite compliance with carbon accounting standards (or at least use standard units and methods). The GHG Protocol’s scope definitions have been explicitly referenced in academic work to

clarify what is included in AI emission calculations. By doing so, a paper can state it calculated emissions "in accordance with Scope 2 accounting", signaling that only electricity use was counted and using a location-based emission factor from an authoritative source (e.g., EPA or IEA data for grid emissions). Similarly, if a study includes manufacturing impact of AI hardware, it may cite LCA standards or prior LCAs of semiconductors to estimate that portion. In short, the community is moving toward standardized reporting: for each new model, report energy (in MWh), carbon emissions (in tons CO_2e with method described), possibly water usage, and assumptions (PUE, emission factor, etc.). This mirrors how other industries report environmental impact and allows comparisons and tracking of improvements over time.

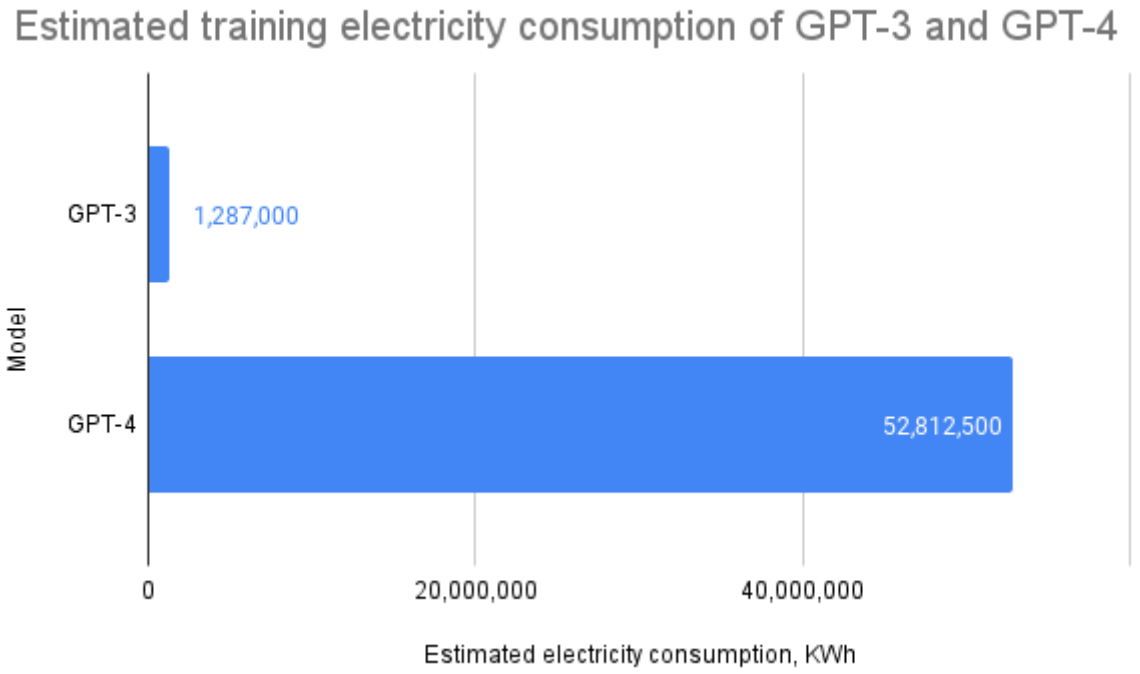
Challenges and Evolving Practices Despite the tools and standards available, measuring LLM environmental impact still faces challenges. One issue is transparency: many AI companies do not disclose full details (e.g., exact energy use, locations, hardware manufacturing data). This makes third-party estimates uncertain. Another challenge is incorporating Scope 3 emissions (like chip manufacturing) reliably, current studies often have to use proxy data or broad assumptions. Nonetheless, the trend in research is to be ever more comprehensive. The BLOOM language model effort in 2022 was noted for providing an extensive environmental impact appendix, covering training energy by region and even inferring the impact of model development (experiments before final training). Following that, the latest works (e.g., Morrison et al. 2025) measure not only training and inference, but also water consumption and encourage using renewable energy or better cooling to mitigate those impacts.

2.3 Carbon Footprint of ChatGPT vs. LLaMA Models

The development and deployment of large language models (LLMs) like OpenAI's ChatGPT (based on GPT-3/GPT-4) and Meta AI's LLaMA family carry a significant carbon footprint. This section compares their energy consumption and CO_2-equivalent emissions, drawing on published metrics for both the one-time training phase and the ongoing inference (serving) phase. We also highlight how differences in model architecture, training methodology, hardware, and infrastructure influence the carbon impact.

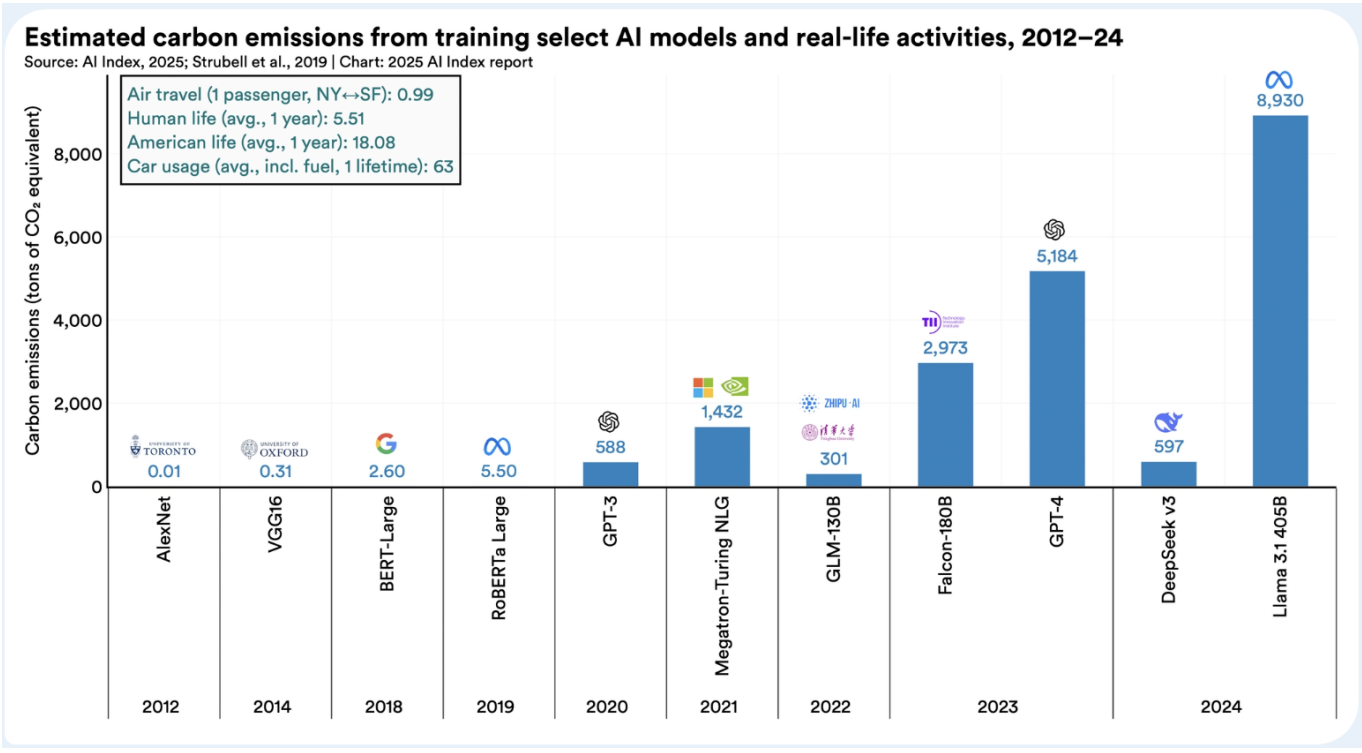
2.3.1 Training Phase: Energy Use and CO_2 Emissions

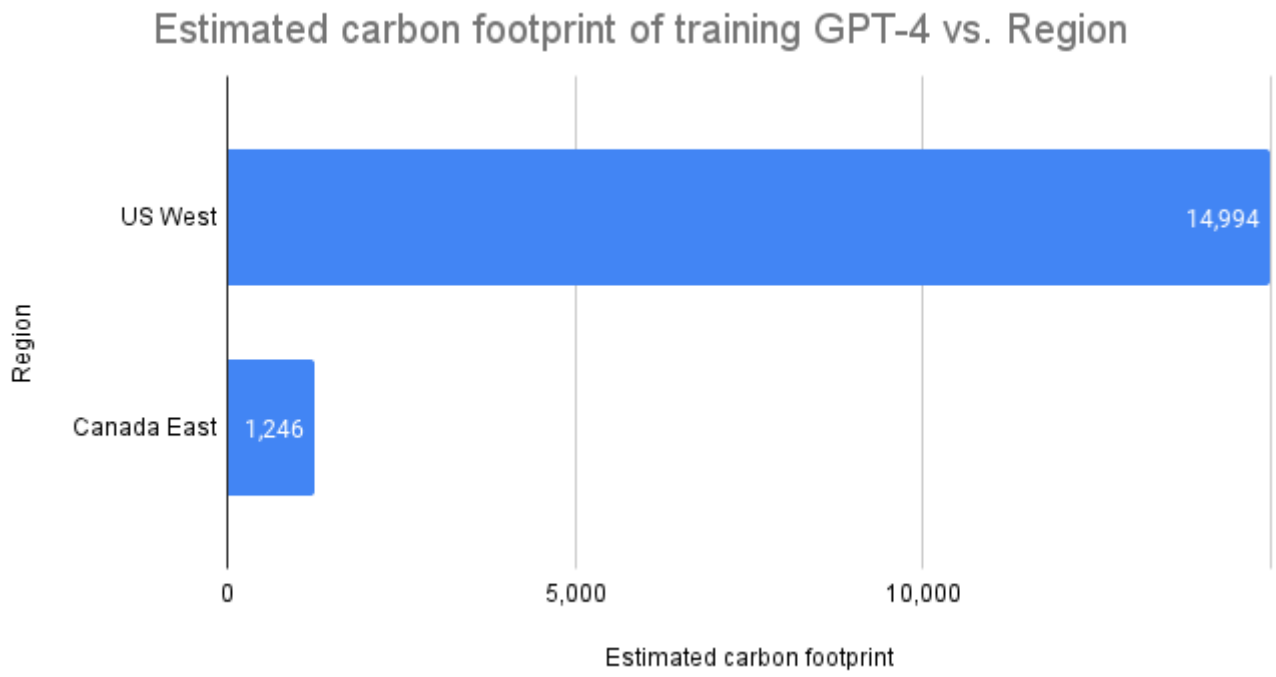
GPT-series (OpenAI) Training state-of-the-art models requires massive computational resources. OpenAI's original GPT-3 model (175 billion parameters, 2020) consumed on the order of ~1.3 GWh of electricity for a single full training run, resulting in an estimated 552 metric tons of CO_2 emissions. This was computed assuming training on cloud GPUs (NVIDIA V100) in a U.S. data center with average grid mix and Power Usage Effectiveness (PUE) near 1.1. The newer GPT-4 model (2023), which is significantly larger and more computationally intensive, required dramatically more energy, on the order of 50–60 GWh of electricity for training.



Graph comparing the estimated electricity consumption from training GPT-3 and GPT-4

Estimates of GPT-4's training footprint range widely based on assumptions about hardware and datacenter efficiency. A recent analysis by Stanford (*Artificial Intelligence Index Report 2025*) reported GPT-4's training emitted ~5,100 tCO_2, roughly 10× the emissions of GPT-3. (Notably, independent calculations suggest it could have been as high as 12,000–15,000 tCO_2 if run on a typical fossil-fueled grid, whereas training on a cleaner-energy datacenter could cut this down to ~1,100 t.)





Graph showing the difference if GPT-4 was trained in the Azure cloud region Canada East its training carbon footprint would have been smaller by a factor of 13.

OpenAI has not publicly disclosed GPT-4's exact energy use or carbon emissions, but these estimates make clear that GPT-4's training phase likely released on the order of thousands of tons of CO_2e. Contributing factors include its larger model size (reportedly an order of magnitude more parameters than GPT-3) and longer training duration. In practice, OpenAI partnered with Microsoft Azure for training; Azure's modern facilities have a low PUE (~1.18) and options for renewable energy, which can mitigate emissions. Still, without full transparency from OpenAI, current figures for GPT-4 remain estimates derived from leaked hardware usage and reasonable assumptions.

LLaMA-series (Meta) In contrast to OpenAI's secrecy, Meta has published detailed carbon accounting for its LLM training. LLaMA-2 (2023) – with model sizes of 7B, 13B, and 70B parameters, was trained on Meta's Research SuperCluster using approximately 3.3 million GPU-hours on NVIDIA A100 80GB GPUs.

Model Size	Time (GPU hours)	Carbon Emitted(tCO2eq)
7B	184320	31.22
13B	368640	62.44
70B	1720320	291.42
Total	3311616	539.00

Meta reports that the total electricity usage for LLaMA-2's training corresponds to ~539 tCO_2e emissions. This figure encompasses all LLaMA-2 model variants and was 100% offset by Meta's sustainability program (Meta purchased renewable energy or credits to neutralize these emissions) Notably, LLaMA-2's carbon footprint is of the same order as GPT-3's, despite LLaMA-2 having fewer parameters (70B vs 175B). This is due to its training on an extremely large dataset of 2 trillion tokens. (By comparison, GPT-3 was trained on ~300 billion tokens.) The larger training corpus for LLaMA-2 increased the compute requirements, effectively offsetting the advantages of its smaller size. However, Meta's use of efficient hardware (A100

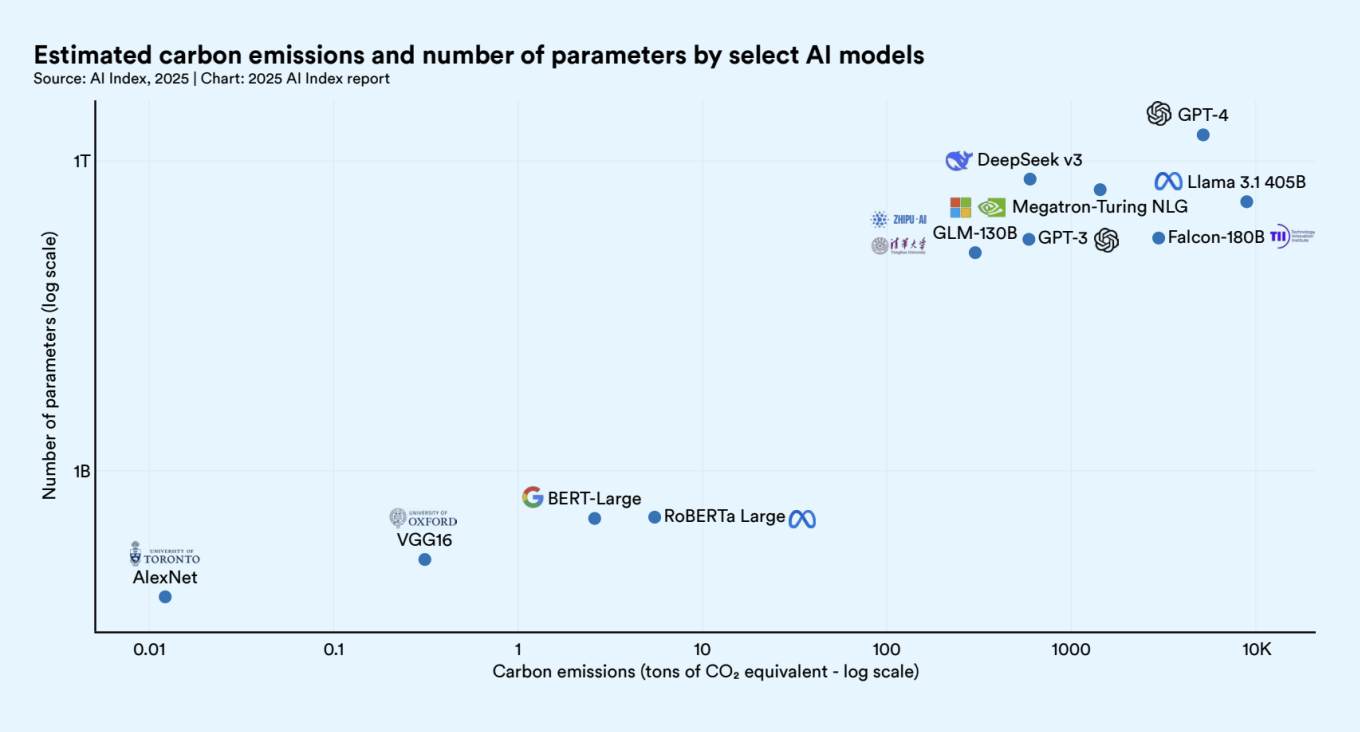
GPUs are more energy-efficient than the older V100s used for GPT-3) and a modern datacenter powered by 100% renewable energy matching helped contain the net emissions. Meta's transparency stands in contrast to OpenAI's approach: the LLaMA-2 model card explicitly lists the training compute and emissions, and emphasizes that others can use the open model rather than re-train new models from scratch, preventing duplicate carbon costs.

Scaling to newer models Both organizations scaled up their models in 2023–2024, with significant implications for carbon footprint. OpenAI's GPT-4, as noted, likely emitted several thousand tons of CO_2. Meta's follow-up LLaMA-3 (2024) was also compute-intensive. While Meta had not publicly released LLaMA-3's full details as of this writing, analyses suggest its training emissions were substantially higher than LLaMA-2. In fact, the Stanford AI Index 2025 reports a 405B-parameter "LLaMA 3.1" model with an estimated 8,930 tCO_2 from training, a reflection of how dramatically emissions rise with model size. This would make LLaMA-3's largest variant about 1.7× more carbon-intensive than GPT-4's training. Even a more modest version of LLaMA-3 (in the 70B parameter range) was estimated to emit roughly 4× the CO_2 of GPT-3.

Image showing the comparison of the carbon footprint of one passenger flight from New York to San Francisco with the carbon footprint of training LLaMA-3, GPT-3 and the average car lifetime emissions.

Meta has indicated it continues to offset 100% of training emissions for its models, and it benefits from a strategy of running data centers on renewable power since 2020 (achieving net-zero operations). By contrast, OpenAI leverages Azure's sustainability initiatives; Microsoft's cloud aims for 100% renewable energy by 2025, which should help reduce the effective carbon intensity of training and serving models. Still, the absolute energy demand of cutting-edge models remains enormous. In summary, GPT-3 and

LLaMA-2 each consumed on the order of 500–600 tCO_2 for training, whereas GPT-4 and LLaMA-3 pushed into the thousands of tons, underscoring the exponential increase in carbon footprint with model scale.



Carbon emissions from training various AI models (in metric tons CO_2e). Early models like AlexNet (2012) were negligible in emissions, whereas modern LLMs are several orders of magnitude higher.

2.3.2 Inference Phase: Energy Consumption and Life-Cycle Impact

While training is a one-time (if expensive) event, the operational energy usage of an LLM during deployment (inference) can dominate its lifetime carbon footprint. Serving millions of user queries on ChatGPT or similar services requires data centers running clusters of GPUs 24/7, ready to generate text on demand. Studies by Meta, AWS, and Google indicate that 60–90% of an LLM's total life-cycle emissions often come from inference usage, not training (*"Carbon Emissions and Large Neural Network Training"*). In other words, a model that is heavily used will burn far more energy over its deployed life than during its initial training. For ChatGPT, which reached 100+ million users, the aggregate computational load is enormous. Each query to ChatGPT involves running the model's forward pass on specialized hardware. OpenAI's public statements note that they "work closely with Microsoft to improve efficiency and footprint" in running these models, but the scale of usage means emissions add up rapidly. A recent analysis estimated that each ChatGPT prompt (query) consumes roughly 4.3 grams of CO_2 on average. This is an order of magnitude more per query than a typical Google web search (~0.2 g CO_2), due to the greater computing required to generate several paragraphs of text. Although 4–5 grams of CO_2 per chat query may sound small, one must consider the volume: for example, 1 million queries would correspond to ~4.3 tons of CO_2. Indeed, at ChatGPT's global usage scale, the model may be responsible for tens of tons of CO_2 emissions per day from inference alone.

Architecture and hardware factors The energy cost of inference is highly sensitive to model size, model design, and the hardware/platform optimizations. OpenAI's latest GPT-4 model (which powers ChatGPT's most advanced version) is believed to have hundreds of billions of parameters, making it computationally heavy for each inference. By contrast, Meta's LLaMA family often emphasizes smaller models (e.g. 7B or 13B parameters for certain use-cases) or efficient architectures, which can be deployed at lower cost. In practice, using a smaller model or more efficient hardware drastically cuts per-query energy. For example,

running a 70B-parameter model on a current NVIDIA A100 GPU in FP16 precision consumes on the order of 15 milligrams of CO_2 per output token generated.

Model scale	Hardware + precision	Carbon per output token
288+ B params	NVIDIA H100 @ FP16	~30 mg CO_2 (estimated for models like GPT-o3 and Llama 4 Behemoth)
70 B params	NVIDIA A100 @ FP16	~15 mg CO_2
70 B params	NVIDIA H100 @ FP8	~7.5 mg CO_2 (≈ 2 × better;)
70 B params	Google TPU v5e @ INT8	~3 mg CO_2 (TPU v5e launch)
13-27 B params	NVIDIA A100 @ FP16	~3 mg CO_2 (applicable to models like Gemma 3 and LLaMA variants)
2 B params	NVIDIA A100 @ FP16	~0.5 mg CO_2

If we instead use a model one-fifth that size (e.g. ~13B parameters) on the same hardware, the carbon per token is about 3 mg, a 5× reduction. Advanced hardware and optimizations can further improve this: the NVIDIA H100 (2022) offers ~2× better efficiency than the A100, and techniques like 8-bit quantization can cut the energy per token by half again. For instance, a 70B model running on an H100 with INT8/FP8 precision might emit only ~7.5 mg of CO_2 per token; Google's TPU v5e, using int8 precision, has been reported to generate as little as ~3 mg CO_2 per token for similar model sizes. These differences mean that Meta's LLaMA models, when deployed at smaller scale or with optimization (e.g. quantization, distillation), can be significantly more carbon-efficient in serving. In fact, Meta has explored model distillation and mixture-of-experts (MoE) architectures for efficiency: an MoE model effectively activates only subsets of the network's parameters for a given query, which can reduce the required compute per inference. OpenAI's GPT-4 architecture is not publicly detailed, but it is suspected to be a dense model running fully for each prompt; thus, its per-token energy use is inherently high. This puts a premium on scalable infrastructure: OpenAI serves GPT-4 via Azure data centers with thousands of GPUs, whereas an open model like LLaMA-2 can be deployed by third-parties on smaller clusters or even on single machines (for the smaller 7B/13B versions), potentially with higher utilization efficiency for specific tasks.

Infrastructure and deployment differences Another factor in carbon impact is where and how the models are hosted. Microsoft Azure (hosting ChatGPT) and Meta's own facilities both boast energy-efficient data centers with high cooling and power efficiency (PUE ~1.1–1.2) and increasing use of renewable energy sources. Meta achieved 100% renewable energy matching for its global operations in 2020, meaning that the electricity used for LLaMA training and inference is effectively compensated with renewables. Microsoft has similarly committed to 100% renewable energy for Azure by 2025. These measures reduce the carbon intensity (kg CO_2 per kWh) of the electricity powering the GPUs. For example, if a model is served from a data center in a coal-heavy region, its emissions per query will be much higher than if served from a region powered by hydro or solar. OpenAI's GPT-4 training analysis showed a 13× difference in CO_2 emissions depending on training location (West US vs. Eastern Canada) due to grid cleanliness. The same principle

applies to inference: deploying models in regions or facilities with cleaner energy can dramatically cut operational emissions. Both OpenAI and Meta appear aware of this; however, transparency differs. OpenAI has not released real-time data on ChatGPT's energy consumption or carbon footprint, and observers have relied on external calculations (and Microsoft's cloud sustainability reports) to estimate its impact. Meta, on the other hand, publishes sustainability reports and included emissions info in LLaMA model cards, explicitly acknowledging the environmental cost and offsets.

ChatGPT and Meta's LLaMA exemplify two approaches to large-scale AI deployment, one as a closed API service and the other as an open model family, but both face the reality that inference energy use eclipses training over the long run. ChatGPT's immense popularity translates into a significant ongoing carbon footprint, partially mitigated by efficient hardware and cloud infrastructure, yet still a cause for concern in aggregate (on the order of thousands of tons of CO_2 yearly for heavy usage). LLaMA-based models, by being open-source, enable users to opt for smaller, task-specific models that are cheaper and greener to run. For instance, an organization could fine-tune a 7B or 13B LLaMA-2 model for a particular application, achieving similar accuracy to GPT-3.5 on that task at a ~10× lower energy cost in inference. Such strategies – model right-sizing, efficient hardware utilization, and renewable-powered deployment are increasingly important for reducing the carbon footprint of AI. In conclusion, ChatGPT (GPT-3/4) and LLaMA (2/3) both require substantial energy, but Meta's models have an edge in transparency and potential for community-led efficiency improvements. The carbon impact of these LLMs can be measured in thousands of tons of CO_2, so optimizing both the training process and the serving infrastructure (e.g. using cleaner energy, better cooling, and more efficient model designs) is critical to making large-scale AI more sustainable. All available data reinforce that bigger models come with disproportionately higher emissions, and going forward, researchers are challenged to "green" the AI lifecycle by adopting the best practices (efficient models, hardware, and hosting) to bend the emissions curve even as model capabilities grow.

2.4 Reducing the Carbon Footprint of Large Language Models

Having quantified the significant emissions of various LLMs in Section 2.3, we now explore strategies to mitigate their carbon footprint without sacrificing performance. Recent research and industry practices suggest multiple approaches to make both the training and deployment of LLMs more sustainable. This section introduces new elements and hypotheses for reducing emissions.

2.4.1 Model Efficiency and Compression Techniques

One promising direction is to make models smaller and more efficient through compression techniques. For example, knowledge distillation can transfer the intelligence of a large model into a smaller one, as demonstrated by DistilBERT's success in retaining most of BERT's accuracy with roughly 40% fewer parameters. By training a compact "student" model to mimic a large "teacher" model's outputs, we achieve comparable performance with less computation, thereby cutting energy usage. Similarly, model pruning removes redundant weights or neurons from a network; research shows that pruning BERT or similar transformers can significantly reduce model size and energy consumption while maintaining accuracy. Another approach, parameter sharing, is used in models like ALBERT, which reuse weights across layers to drastically shrink the model's memory footprint and accelerate training. These methods highlight that reducing the number of parameters (or operations) directly translates to lower carbon emissions, since there are fewer computations to power.

Beyond structural compression, numerical efficiency can be improved through low-precision arithmetic. Quantization involves using lower-bit representations (e.g. 8-bit integers instead of 32-bit floats) for model weights/activations, which has been shown to significantly lower the energy cost of both training and inference, especially for large models . By quantizing a model, we reduce memory usage and increase hardware throughput, so the same task requires less electricity. Notably, one study found that 8-bit quantization can cut the inference carbon emissions of a large model substantially without degrading its output quality, making it a practical way to deploy LLMs more sustainably . In tandem with quantization, researchers are also exploring mixed precision training (using half-precision floats) and novel training algorithms that find good solutions with fewer iterations or less exact computations. For instance, early-stopping techniques (halting training once validation metrics plateau) and efficient optimization methods can avoid needless epochs – thereby saving energy when additional training yields diminishing returns. Overall, smarter training protocols and compressed model architectures represent a key frontier in reducing LLM footprints: they attack the problem at its source by requiring the model to do less work for the same result.

2.4.2 Hardware and Infrastructure Optimizations

Another major avenue for reduction is leveraging more efficient hardware and data center infrastructure. Modern AI accelerators (such as the latest GPUs or TPUs) can perform vastly more computations per watt of energy than older hardware. Empirical results demonstrate that upgrading from an older GPU (NVIDIA T4) to a newer one (A100) for LLM training cuts both training time and emissions drastically. In one experiment, switching to the A100 reduced training time by ~62% and lowered CO2 output by 83% on average for the same set of models and tasks. Importantly, this gain did not come at the cost of model performance faster chips simply achieve the result more efficiently. Thus, utilizing high-performance, energy-efficient hardware is a direct way to curb emissions. Specialized AI chips and GPUs (versus only CPUs) also help, as they deliver more computation per unit of energy. Additionally, advanced cooling solutions in hardware (e.g. liquid cooling for GPU servers) can reduce the power overhead for cooling, further improving the overall efficiency of the training setup.

That said, there is a trade-off to consider with new hardware: the embodied carbon from manufacturing these cutting-edge devices. Producing a top-tier GPU carries a significant carbon cost, which can offset some benefits if hardware is refreshed too frequently. A sustainable strategy is to extend the lifespan of existing hardware while operating it efficiently. For example, researchers at MIT Lincoln Lab found that by capping the power draw of GPUs (running them at slightly lower peak wattage), they could reduce energy use by ~15% for training jobs at the cost of only ~3% longer runtimes.

"We studied the effects of capping power and found that we could reduce energy consumption by about 12 percent to 15 percent, depending on the model," Siddharth Samsi

This small slowdown was *“barely noticeable”* in multi-day trainings, yet it kept the GPUs running ~30°F cooler and eased strain on cooling systems. Cooler, less stressed hardware is less prone to failures and can remain in service longer, delaying the need for new hardware purchases (and the associated manufacturing emissions). In summary, optimizing hardware usage whether by choosing energy-efficient processors or tuning their operation can yield large emission savings. Companies and research labs are encouraged to invest in “green computing” infrastructure, as these improvements often pay dividends both environmentally and financially (through energy cost savings and hardware longevity).

2.4.3 Carbon-Aware Computing Practices

Even with efficient models and hardware, when and where we perform LLM training can influence its carbon footprint. A growing body of work advocates for carbon-aware computing, which aligns heavy computations with times and locations that have cleaner energy. One straightforward step is using renewable energy sources to power AI data centers. If a training run is executed in a region or facility powered by hydro, solar, wind, or nuclear energy, the net CO2 emissions drop dramatically (potentially near-zero if fully renewable). Several cloud providers now let users select low-carbon data center regions, and some AI initiatives (such as the BLOOM model's training) leveraged France's nuclear-heavy grid to keep emissions low. When direct use of clean power isn't possible, purchasing renewable energy credits or carbon offsets is another way organizations neutralize the impact of electricity used in training (though offsets are a secondary resort to actual emission reduction).

Another technique is scheduling AI workloads based on grid and cooling conditions. For instance, one study proposed pausing training jobs when a region's electricity is coming mostly from high-carbon sources (e.g. during peak demand on a coal-heavy grid) and resuming when the power mix improves or demand drops. In a long-running job, this adaptive scheduling yielded up to ~25% reduction in emissions without requiring any changes to the model. Likewise, data centers can time-intensive jobs for cooler nighttime hours or winter months, reducing the energy needed for cooling servers. Many of these carbon-aware strategies are just beginning to be tested in practice, but early results indicate that substantial savings are possible by simply being mindful of when we train. The overhead in training time is often minor relative to the environmental benefit for example, pausing a multi-week run during a few peak hours might extend total training by a day, but avoid a large amount of peak-hour emissions. As tools for real-time carbon intensity data become more available, we can expect such smart scheduling to become easier and more common.

In summary, by combining clean energy, intelligent scheduling, and efficient facility design, we can significantly reduce the carbon emissions associated with running large models on existing hardware.

2.4.4 Reusing Models and Emerging Approaches

A complementary way to reduce the overall environmental impact of LLM development is to avoid unnecessary retraining. Instead of training new models from scratch for every task or organization, the field is moving toward reusing and fine-tuning pre-trained models. Using an existing LLM as a starting point (especially an open-access model with known emissions) and fine-tuning it for a new task requires only a fraction of the computational effort compared to full training. This approach not only saves time and resources but also acts as a form of "emissions recycling", the large one-time cost of training a GPT-3 or BLOOM can be amortized over hundreds of derived applications. Encouraging model sharing and avoiding duplicate large trainings is therefore a pragmatic way to curb the collective carbon footprint of the AI community. For example, if one group has already trained a 10-billion-parameter model on a huge dataset, other groups can build on those weights instead of burning compute to train a similar model from scratch. Initiatives like Hugging Face's model hub facilitate this by making pre-trained models widely available for reuse.

In addition, transparency in reporting the energy usage and emissions of training runs is crucial. By openly documenting the carbon cost of models, researchers can create accountability and track progress as techniques improve. The BLOOM project is a case in point: it published a detailed carbon footprint analysis (estimating ~25 to 50 tonnes CO2 for the full training) and discussed the methodology used to minimize and offset those emissions. Such reporting not only highlights the problem to the community but also allows comparison and competition on energy efficiency, not just accuracy. Over time, this could incentivize more innovations for energy reduction (much as model size or speed optimizations are incentivized today).

Finally, looking forward, there are emerging technologies that could dramatically alter the energy paradigm of AI. Research into neuromorphic computing (brain-inspired chips) and analog AI accelerators promises orders-of-magnitude improvements in efficiency by changing how calculations are done at the hardware level. Likewise, exploring algorithms like sparse models or mixture-of-experts that activate only portions of the model as needed can reduce the active compute per query. While these ideas remain largely experimental or theoretical for now, early trials have shown potential in limited domains. For instance, mixture-of-experts models have achieved comparable results to dense LLMs while using less computational power by routing each input through a small subset of “expert” parameters instead of the entire network. As these approaches mature, they could open up new ways to maintain high performance with significantly lower energy consumption essentially hypotheses being tested on a few examples today that might become mainstream tomorrow.

In summary, reducing the carbon footprint of LLMs will likely require a combination of the above strategies. By building smaller or more efficient models (through distillation, pruning, and quantization), leveraging efficient hardware (while managing its full life-cycle impact), and adopting carbon-aware practices (clean energy, smart scheduling, and model reuse), it is possible to substantially cut emissions without diminishing AI capabilities. Encouragingly, studies have shown that many of these measures can be implemented with minimal impact on model accuracy or training outcomes. The challenge ahead lies in scaling up these solutions and integrating them into standard AI development pipelines. Embracing such sustainable AI practices is a necessary step to ensure that future LLM breakthroughs do not come at the expense of the environment. By innovating in how we design, train, and deploy models, we can drive down the environmental cost of AI even as we push the boundaries of model performance.

2.5 Leveraging Large Language Models to Reduce Carbon Footprints

2.5.1 Energy Management and Grid Operations

While Section 2.4 examined how to curb the environmental cost of running LLMs, this section investigates how the models themselves can be used as tools for decarbonization.

Recent peer-reviewed and preprint studies demonstrate that large language models (LLMs) can actively contribute to decarbonization by improving forecasting, modeling, and demand-response participation. In the power sector, *Ma et al.* (2025) designed a multi-agent interactive load-forecasting framework in which an LLM orchestrates specialised agents for data cleaning, forecast generation, and human feedback. Their experiments showed that forecast accuracy improved when system operators supplied contextual insights —such as upcoming holidays or weather anomalies—and that the framework remained affordable for real-world deployment. For instance, during test runs the system allowed operators to inform the model about an impending heat wave; the LLM incorporated that information, producing a more accurate demand forecast and enabling dispatchers to rely on cleaner generation rather than firing up a gas peaker plant. Such human-in-the-loop forecasting illustrates how LLMs can both lower the barrier for non-expert use and reduce carbon-intensive reserves.

On the demand side, *He et al. (2025)* developed an LLM interface for Home Energy Management Systems (HEMS) that transforms free-form user descriptions—like “we cook dinner around 6 p.m. and do laundry on weekends”—into eight well-structured control parameters. The system uses a ReAct (reason-and-act)

prompting strategy with few-shot examples, significantly raising extraction accuracy. In their evaluation, the Mistral-7B-Instruct-v0.2 model with ReAct+example prompts achieved 96.9 % accuracy on easy tasks, 87.5 % on medium, and 78.8 % on hard tasks. A practical demonstration involved a household whose electricity bill dropped from £31 to £16 (a 48 % reduction) when the LLM-enabled HEMS scheduled appliances according to real-time electricity prices. This example underscores how LLMs can dramatically increase participation in demand-response programs, unlocking flexible loads that were previously underused.

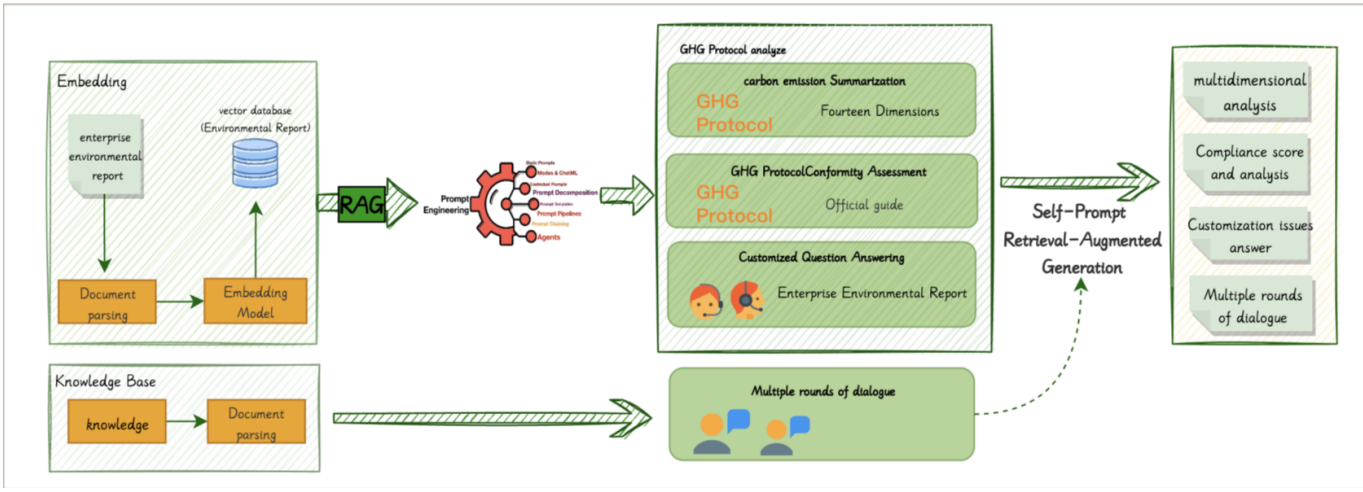
Building operations also offer fertile ground for LLM applications. *Gao et al. (2024)* integrated ChatGPT with the EnergyPlus simulation engine, showing that the model can automate simulation input generation, analyse outputs, detect anomalies, and even suggest design adjustments. In one case study, a natural-language description of a modular apartment building ("the iUnit") was converted by a multi-agent LLM into a complete EnergyPlus input data file (IDF) containing dozens of object definitions; after debugging by the LLM agents, the file ran without errors and matched baseline simulation results. This automated workflow reduced weeks of manual IDF scripting to a matter of hours. Jiang et al. (2025) further argued that LLMs can support automated energy-model generation, fault detection and diagnosis, and real-time energy-management optimisation, and recommended research directions such as domain-specific fine-tuning, retrieval-augmented generation (RAG), and multimodal integration to handle the complexity of building energy data. As a result, the deployment of LLMs in building energy modeling could enable rapid "what-if" analyses, accelerate retrofitting assessments, and help architects and engineers design low-carbon buildings more efficiently.

In commercial buildings and district energy systems, LLMs are emerging as assistants for energy modeling and optimization. The EnergyPlus–LLM integration reported by *Gao et al.* uses ChatGPT to auto-generate simulation inputs, analyze outputs, and even suggest design alterations. Their case studies show that proper prompt engineering allows the model to produce accurate building-energy simulations and identify anomalies, cutting down the engineering time required for each project. A parallel perspective paper in Building Simulation argues that LLMs can streamline automated energy model generation, fault detection and energy management optimization. Although these studies remain at early stages, they suggest that LLM-driven assistance could accelerate efficiency retrofits and optimize HVAC control, both of which reduce buildings' operational emissions.

2.5.2 Corporate Carbon Accounting and Product Footprints

Beyond optimizing energy operations, large language models (LLMs) can play a pivotal role in measuring and reporting emissions—an essential precursor to effective decarbonization. Two strands of research highlight these opportunities.

The CarbonChat system demonstrates how LLMs can automate complex sustainability reporting tasks. Instead of manually parsing lengthy corporate sustainability reports, CarbonChat uses a retrieval-augmented generation (RAG) architecture to break documents into logical chunks, retrieve relevant sections, and answer tailored queries. Its architecture includes a diversified index module (combining document-tree and semantic chunking) and a self-prompting pipeline that integrates intent recognition, structured chain-of-thought prompting, and hybrid retrieval; this ensures both syntactic and semantic fidelity.



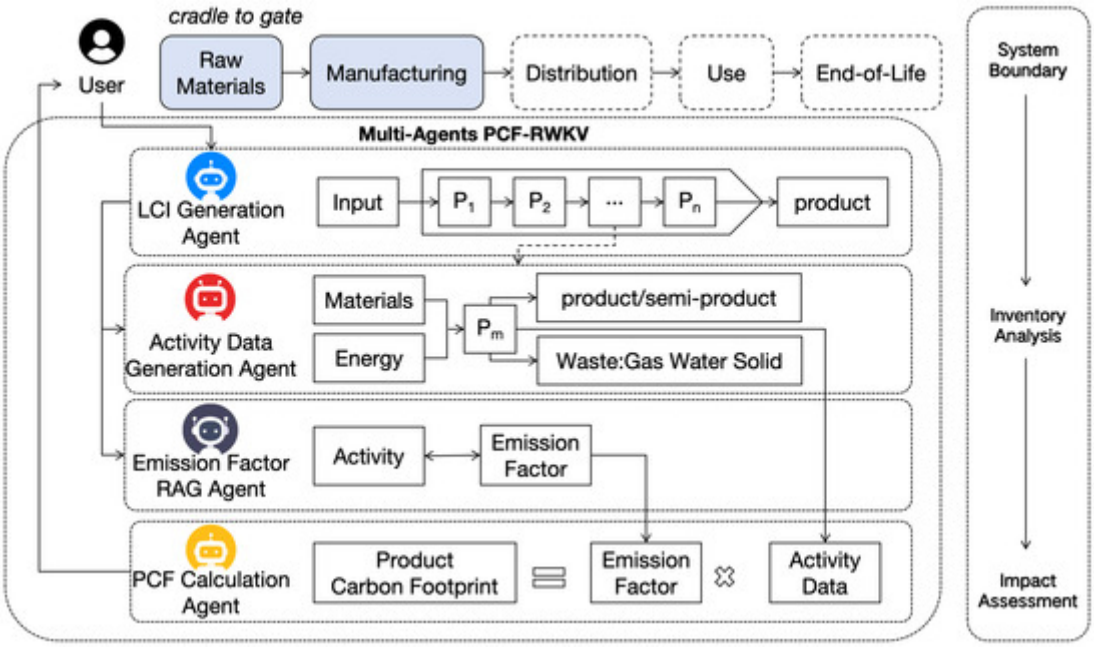
CarbonChat architecture diagram.

CarbonChat then maps information into fourteen analytical dimensions corresponding to Scope 1, Scope 2, and Scope 3 categories in the Greenhouse Gas Protocol. Extensive experiments reported large improvements in retrieval accuracy and response relevance (measured by ROUGE and BERTScore) and high Text2SQL execution accuracy. In practice, companies can use CarbonChat to generate compliant carbon accounting reports and climate-related Q&A in minutes rather than the days of manual effort often required by sustainability analysts.

A parallel effort at the Climate Change AI workshop developed a multi-agent architecture that combines an LLM with retrieval and classification modules to systematically extract carbon-reduction actions from corporate reports. Although detailed results have yet to be published, the framework uses specialized agents for parsing, classification, and data integration; it builds a sector- and region-specific library of mitigation levers and updates as new reports are published. Such a system could help sustainability teams rapidly identify actionable opportunities (e.g., switching to low-carbon fuels, investing in energy efficiency) across multiple firms and geographies.

At the product level, traditional life-cycle assessments require painstaking data gathering and boundary setting. *Zhang et al. (2024)* propose a Product Carbon Footprint RWKV (PCF-RWKV) model that automates much of this work. Their system employs a multi-agent architecture comprising four key agents:

- The LCI Generation Agent builds a product-specific life-cycle inventory by analysing material and energy flows at every stage of the product's life.
- The Activity Data Generation Agent converts the LCI into detailed activity data (e.g., energy consumption, waste outputs) for each process step.
- The Emission-Factor RAG Agent uses search-enhanced generation to retrieve or calculate appropriate emission factors from a corpus of life-cycle assessment documents.
- The PCF Calculation Agent combines activity data and emission factors, using an LLM to generate and execute carbon-footprint calculations tailored to the specific product.



PCF-RWKV multi-agent architecture diagram.

An orchestration agent coordinates these steps: it refines user queries, retrieves literature, and ensures each stage feeds into the next. Because the PCF-RWKV model employs a low-rank adaptation (LoRA) of the RWKV architecture and runs on a single consumer-grade GPU, it avoids the carbon emissions associated with cloud-based high-performance computing. The authors report that their design can generate reliable "cradle-to-gate" footprints and that its emission-factor RAG agent dynamically updates factors to reflect the latest research, improving accuracy over static databases.

By transforming unstructured sustainability reports into actionable emission profiles and automating life-cycle assessments through multi-agent LLM architectures, these examples show that LLMs can drastically reduce the time, cost, and expertise needed for robust carbon accounting. Such tools have the potential to lower the barriers for companies—especially small and medium enterprises—to measure and manage their emissions, thereby accelerating progress toward emissions reduction targets.

2.5.3 Climate Information and Decision Support

Ensuring that communities, businesses and policymakers have easy access to reliable climate and energy information is a prerequisite for effective decarbonization. Recent work suggests that large language models (LLMs) can fulfil this role by summarizing, contextualizing and communicating complex climate data.

A 2024 study in Communications Earth & Environment introduced ClimSight, an LLM-based climate-service prototype. The authors showed that LLMs can integrate heterogeneous datasets—geographical information (land use, soil type, distance to the coast), historical climate observations and future climate projections—to answer location-specific questions about climate change. For example, a user in Morocco asked: "I intend to cultivate wheat. What are the implications of climate change?". ClimSight retrieved global climate model output for the user's coordinates (1985–2004 and 2070–2100), compared baseline and future temperature, precipitation and wind patterns against wheat's optimal growing conditions, and delivered a structured response. The output noted that rainfall in the region is projected to decrease by 11–20 %, with summers becoming hotter by 1.9–5.4 °C and winds intensifying by up to 0.38 m/s. It advised the farmer to invest in irrigation, adopt heat-resistant wheat varieties, protect crops from higher winds, and monitor local climate policies. Notably, each query cost approximately €0.06, suggesting that such personalized climate services

could scale affordably. This example illustrates how fine-tuned LLMs can transform abstract climate projections into actionable guidance for specific locations and sectors.

Commercial efforts are also exploring climate-focused LLMs. ClimateGPT, developed by AppTek, EQTYLab and Erasmus AI, was fine-tuned on roughly 300 billion climate-related tokens and subjected to multilingual instruction tuning. The developers market it as a cross-lingual "climate social intelligence" engine that can answer questions about climate science, policy and finance in several languages—an indication that targeted training and retrieval can turn a general model into a specialized environmental assistant. Although the model's performance has not yet been peer-reviewed, it demonstrates an industry push to provide policymakers, researchers and businesses with accessible, domain-specific climate knowledge.

Decarbonizing small and medium enterprises (SMEs) requires digesting a proliferation of policies, incentives and technical options. To meet this need, *Arslan et al. (2024)* designed an Energy Chatbot that combines LLMs with a multi-source retrieval-augmented generation framework. The system indexes news articles, government documents, academic research, and social media to answer questions about sustainable energy initiatives, financial incentives and regulatory changes. In a demonstration scenario, an SME in the hospitality sector queried the chatbot about available subsidies for installing solar panels and incentives for adopting heat-pump technology. The bot drew on recent government climate-action plans and industry newsletters, returning a concise summary of grants, tax credits, and expected payback periods. By lowering the research burden and presenting information in plain language, the Energy Chatbot enables SMEs to formulate long-term sustainability strategies and reduce energy costs. While full experimental details are not publicly available, the concept underscores how multi-source LLM systems can democratize energy knowledge for organisations lacking in-house expertise.

2.5.4 Making LLMs Themselves More Sustainable

Growing awareness of AI's environmental footprint has spurred studies on how to make LLMs themselves more sustainable. A joint report by UNESCO and University College London shows that relatively simple design choices can dramatically reduce energy use without degrading performance. Their experiments, conducted across multiple open-source LLMs, found that generative models are collectively enormous energy consumers—each prompt to a commercial LLM uses about 0.34 Wh of electricity, and the more than 1 billion daily interactions equate to ≈ 310 GWh per year, similar to the annual electricity consumption of over 3 million people in a low-income African country. To mitigate this impact, the report recommends three key strategies:

1. **Smaller, task-specific models**: The team observed that lean models tailored to specific jobs (e.g., translation or summarization) can achieve comparable accuracy to general-purpose giants, while cutting energy use by up to 90 %. This "right model for the right job" approach challenges the prevailing tendency to use one large model for all tasks; instead, a translation task should call a specialised translation model, and summarisation should use a summariser. Such modularity reduces idle parameter activity and thus energy consumption. The report also highlights mixture-of-experts (MoE) architectures, where a single system contains many specialised sub-models and activates only those needed for a given task. By routing each input through a small subset of "experts," MoEs maintain performance while curbing unnecessary computation.
2. **Concise prompting**: UCL's experiments show that shorter prompts and responses can reduce energy use by over 50 %. Each additional token processed requires matrix multiplications across billions of parameters, so trimming superfluous context yields significant savings. For example, simply removing filler phrases ("please summarise the following text") and using direct instructions

("summarise:" + text) reduces both latency and power draw. The researchers argue that concise prompting should become a standard efficiency practice, akin to code optimisation.

3. **Model compression**: Techniques such as quantisation and pruning—collectively known as model compression—reduce the number of bits or parameters used to represent the model. According to the report, compression can save up to 44 % in energy while preserving accuracy. This aligns with Section 2.4's discussion of compression methods like Low-Rank Adaptation (LoRA) and underscores their utility not only for speed but also for sustainability. Importantly, the report emphasises that adopting smaller models, MoE architectures and concise prompts makes AI more accessible in low-resource settings, where computational power and energy are scarce; only 5 % of Africa's AI talent currently has access to sufficient computing resources.

Complementary to design changes, deployment strategies can mitigate AI's carbon footprint. A 2025 meta-analysis of 390 generative AI models estimates that these models consume 24.97–41.10 TWh of electricity and emit 10.67–18.61 million tons of CO_2 over the 2018–2024 period. Notably, the United States and China together account for 99 % of the emissions, while Europe accounts for only 0.02–0.09 million tons. The study concludes that relocating model training and inference to regions with lower grid carbon intensity—for instance, Scandinavia or the UK, which have higher shares of renewable electricity—could substantially lower emissions. In addition to energy, the analysis warns of accumulating e-waste: projections suggest that GAI infrastructure could generate 16 million tons of electronic waste by 2030. These findings highlight that even as hardware becomes more efficient, geographical placement of data centres and adoption of renewable energy sources remain critical factors in AI sustainability.

Beyond generic design and deployment strategies, it is instructive to see how leading developers are already moving toward right-sizing AI workloads. In August 2025, OpenAI introduced GPT-5, describing it as a "unified system" composed of three components: (i) a smart, efficient model for most questions, (ii) a deeper reasoning model for more complex tasks ("GPT-5 thinking"), and (iii) a real-time router that decides which model to use based on the conversation's complexity, the tools needed, and explicit user intent. The router is continually trained on real usage signals (e.g., when users switch models or rate responses), enabling it to improve its routing decisions over time.

While OpenAI's announcement does not explicitly mention energy, this architecture reflects the principle of dynamic model selection advocated by sustainability experts: by routing simple queries to a smaller "efficient" model and reserving the larger "thinking" model for genuinely complex tasks, the system avoids expending unnecessary computational resources. This aligns with the recommendations of UNESCO and UCL (discussed earlier) to use smaller task-specific models and mixture-of-experts designs to cut energy consumption by up to 90 %. In other words, GPT-5's built-in router can be viewed as an operational manifestation of these guidelines—one that could materially reduce the energy footprint of everyday interactions by matching the model size to the task at hand. The introduction of "mini versions" for overflow queries further exemplifies how resource-aware model tiers might help cap the total compute used per user. Such innovations underscore that sustainable AI need not be limited to offline optimisation; intelligent orchestration and model selection at runtime can also play a pivotal role in keeping large language models' energy usage in check.

Conclusion

This thesis set out to interrogate the environmental consequences of large language models (LLMs) and to explore how these same systems might advance climate mitigation. Through an examination of their

architectural origins and explosive growth, we traced how the extraordinary scale that enables LLMs to translate, reason and summarise simultaneously inflates their energy and carbon demands. Applying established carbon-accounting standards and life-cycle metrics allowed us to quantify those demands and to reveal how model size, training data and deployment choices determine the overall footprint. By comparing the environmental profiles of representative systems from OpenAI and Meta, we demonstrated that increases in model parameters and training tokens drive emissions up non-linearly, while transparency and renewable energy procurement can meaningfully reduce them. We then synthesised a suite of mitigation strategies—spanning model compression, efficient hardware, carbon-aware scheduling and re-use of pre-trained weights—that collectively show how researchers and practitioners can curb the energy cost of AI without diminishing its capabilities. Finally, recognising that LLMs are not only consumers of energy but also potential instruments of sustainability, we showcased emerging applications of these models in energy management, carbon accounting and climate communication, illustrating that they can help reduce emissions even as they generate them. Together, these strands of evidence respond to the guiding question and illuminate both the risks and the opportunities posed by LLMs in an era of urgent climate action.

In my view, the advent of LLMs represents a societal transformation on the scale of the nineteenth-century industrial revolution. These models are becoming embedded in education, communication, commerce and infrastructure; they promise enormous productivity gains and new forms of knowledge creation. Yet our modern context is characterised by climate urgency. We therefore cannot afford to repeat the mistakes of early industrialisation, which externalised environmental costs for decades. LLM development and deployment must not add to our ecological debt.

Two broad principles follow from this thesis. First, not every problem requires a giant foundation model. Many tasks can be solved with smaller, task-specific networks or retrieval-based systems that use orders of magnitude less energy. Moreover, prompt length alone can halve electricity consumption. Reserving large LLMs for challenges that genuinely require deep contextual reasoning will avoid unnecessary emissions.

Second, the physical infrastructure that hosts and serves LLMs must be decarbonised. Today’s data centres are unevenly optimised; some rely heavily on fossil-fuelled grids or fail to recover waste heat. The German Energy Efficiency Act provides a blueprint for regulatory action: new data centres operating from July 2026 must reuse at least 10 % of their energy (e.g., by supplying heat to district heating networks), rising to 20 % by July 2028; they must source 50 % of their electricity from renewables from 2024 and 100 % from 2027; and they must implement certified energy or environmental management systems by mid-2025. Similar requirements appear in an Energy Reuse Factor (ERF) mandate and PUE thresholds for new data centres. Such rules illustrate how governments can mandate renewable sourcing, efficiency improvements and waste-heat recovery while imposing meaningful fines for non-compliance. I believe other jurisdictions should adopt comparable standards: siting facilities in cold regions to reduce cooling needs, coupling them with district heating to reuse waste heat, and powering them with wind, hydro or solar energy.

Achieving climate-compatible AI will require a broad coalition. Technology companies need to embed energy and emissions accounting into their product cycles, transparently reporting the footprint of model training and deployment. Regulators should enforce minimum efficiency standards, renewable procurement and waste-heat reuse for new data centres, as Germany has begun to do. Users and developers must cultivate a culture of “model sufficiency”, resisting the instinct to solve every problem with the largest available model. Researchers should continue to explore hybrid architectures (such as diffusion-based LLaDA and LLDM models, which combine autoregressive language modelling with diffusion processes) and

neuromorphic chips that promise orders-of-magnitude improvements in efficiency. Finally, the sustainability community must ensure that policies keep pace with technological innovation—adapting standards, fostering renewable energy integration and addressing e-waste.

The industrial revolution irrevocably altered human civilisation and the planet's climate. We have an opportunity to guide the AI revolution along a more sustainable path. Large language models can help us mitigate climate change—through better forecasting, optimisation and decision support—but only if we design and deploy them with environmental responsibility at the forefront. Balancing the promise of AI with the imperative of climate action is not optional; it is the defining challenge of our technological era.

Key Academic Sources and White Papers (with Annotations)

Sources: 1. Vaswani, A., et al. (2017). Attention is All You Need. Advances in Neural Information Processing Systems. 2. Devlin, J., et al. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. NAACL. 3. Brown, T., et al. (2020). Language Models are Few-Shot Learners. NeurIPS. 4. Zhao, W. X., et al. (2023). A Survey of Large Language Models. arXiv preprint arXiv:2303.18223 . 5. Khan, A., et al. (2025). Industrial Applications of Large Language Models. Scientific Reports 15, 12345. Mubashar Raza, Zarmina Jahangir, Muhammad Bilal Riaz, Muhammad Jasim Saeed & Muhammad Awais Sattar Scientific Reports volume 15, Article number: 12345 (2025) . 6. Bommasani, R., et al. (2021). On the Opportunities and Risks of Foundation Models. Stanford CRFM. 7. MIT News (2025). Explained: Generative AI's environmental impact .

2.2 Key Academic Sources and Their Relevance

• Strubell et al. (2019) – "Energy and Policy Considerations for Deep Learning in NLP." This influential study from UMass Amherst provides one of the first quantifications of the environmental cost of modern AI models . The authors measured the GPU/CPU energy consumption for training several NLP models and then computed the CO_2 emissions using standard conversion factors. They famously found that a large Transformer with neural architecture search emitted ~626,000 lbs of CO_2 (284 metric tons) . This paper is relevant as it exemplifies how to apply energy measurement tools and emission calculations to AI, and it argues for reporting energy costs in academic research . It establishes baseline methods (power monitoring, use of average emission factors) that are built upon by later works. • Lannelongue et al. (2021) – "Green Algorithms: Quantifying the Carbon Footprint of Computation." An academic article (Advanced Science) led by researchers at Cambridge University, it introduces a general framework and tool for calculating the carbon footprint of any computational task . The authors distilled the problem to a few key parameters: hardware type, runtime, and location (energy grid mix), and provided an open-source online calculator . The paper is highly relevant as it presents a standardized method to estimate energy and emissions, which can be directly applied to AI workloads. It also discusses the importance of transparency and includes strategies to reduce carbon footprint (e.g. using more efficient hardware or greener energy) . This source demonstrates an academic approach to codifying measurement practices and has influenced subsequent tools like CodeCarbon. • Wu et al. (2022) – "Sustainable AI: Environmental Implications, Challenges and Opportunities." A comprehensive survey paper (from Facebook AI Research, presented at MLSys 2022) that examines the environmental impact of AI from a holistic perspective . It spans the entire machine learning lifecycle – data, model training, hardware – and explicitly characterizes the carbon footprint of AI computing, including operational and manufacturing emissions . The paper's relevance lies in its discussion of measurement challenges: it highlights the need for better data on hardware manufacturing and advocates for industry-wide adoption of measurement standards. It also provides insights into optimizing hardware/software for efficiency and calls out the role of lifecycle assessment by examining

hardware production and end-of-life . In our context, this source supports the view that measuring LLM impact requires a broad systems approach and validates the importance of standards like LCA and holistic reporting. • Morrison et al. (2025) – "Holistically Evaluating the Environmental Impact of Creating Language Models." A recent (2025) study by researchers from Allen Institute for AI and CMU (including Emma Strubell) that sets a new bar for detailed LLM impact analysis. This work is notable for applying GHG Protocol standards (Scope 2) in calculating emissions for training a series of LLMs , and for going further to measure previously under-reported aspects: development-phase energy, embodied carbon of hardware, and water usage . They reported that even training relatively small LLMs (up to 13B parameters) in a efficient data center resulted in ~493 tCO_2 and consumed 2.8 million liters of water . The paper's methodology section (using power meters at sub-second intervals, region-specific emission factors, etc.) exemplifies current best practices . This source is highly relevant as it demonstrates how the methods and standards discussed in Section 2.2 are concretely implemented to produce a thorough environmental assessment of LLMs, reinforcing the section's points about comprehensive measurement and transparency. • OECD (2022) – "Measuring the Environmental Impacts of AI: The AI Footprint." An OECD policy paper that, while not solely academic, aggregates research and proposes directions for standardizing AI impact measurement . It distinguishes direct impacts (compute energy, e-waste) from indirect effects and recommends the establishment of measurement standards specific to AI . This report is included for its authoritative overview and its emphasis on looking beyond operational emissions (e.g., considering lifecycle and indirect impacts) . It supports the thesis section by providing a high-level validation that measuring AI's environmental impact is a recognized priority, and it cites many of the academic works above as evidence. The OECD's recommendations underscore why the methods and standards in this outline are critical – to enable policymakers and stakeholders to reliably track the sustainability of AI development.

2.3 Key References and Their Relevance

1. Strubell, Emma, et al. (2019). "Energy and Policy Considerations for Deep Learning in NLP." arXiv:1906.02243. – A seminal study quantifying the environmental cost of training NLP models. Strubell's team found that training a single large transformer could emit over 626,000 pounds of CO_2 (equivalent to five cars' lifetime emissions) . This reference establishes the high carbon footprint of early large models, providing baseline context for why comparing GPT-3/4 and LLaMA models' footprints is important.
2. Schwartz, Roy, et al. (2020). "Green AI." Communications of the ACM 63(12): 54–63. – This article introduced the concept of "Green AI," arguing that AI research should prioritize energy efficiency and carbon footprint alongside accuracy. It highlights the trade-off between "Red AI" (maximizing performance at any cost) and "Green AI" (focusing on computational efficiency). Used in the thesis to underscore the motivation for comparing ChatGPT and LLaMA models' energy usage, and to frame how newer models (like LLaMA) strive for competitive performance with lower resource and emissions costs .
3. Patterson, David, et al. (2021). "Carbon Emissions and Large Neural Network Training." arXiv:2104.10350. – Authored by Google researchers, this study reports the energy consumption and CO_2 emissions of training frontier models. Notably, it estimates OpenAI's GPT-3 (175B) training consumed 1,287 MWh of electricity, producing ~552 metric tons of CO_2 . The authors identify best practices to curb emissions – e.g. using efficient hardware and cleaner energy can cut carbon output by 100×–1000× . This source provided concrete data on ChatGPT's (GPT-3's) footprint and informed comparisons with Meta's models.
4. Luccioni, Sasha, et al. (2022). "Estimating the Carbon Footprint of BLOOM, a 176B Parameter Language Model." arXiv:2211.02001. – Hugging Face researchers measure the full life-cycle

emissions of BLOOM (an open 176B model) and compare them to other LLMs. They report BLOOM's training emitted only 25 tCO_2 (using mainly nuclear energy in France), versus 502 tCO_2 for GPT-3 trained on a fossil-heavy grid . This reference was used to illustrate how energy source and efficiency choices lead LLMs like BLOOM and LLaMA to a much smaller carbon footprint than GPT-3, and to provide published figures for GPT-3's emissions for the thesis comparison.

5. Touvron, Hugo, et al. (2023). "Llama 2: Open Foundation and Fine-Tuned Chat Models." arXiv:2307.09288 (Meta AI). – This technical report introducing LLaMA 2 (7B–70B) includes a section on training energy use. The authors disclose that pretraining LLaMA 2 models required a total of 3.3 million GPU-hours on NVIDIA A100s, resulting in an estimated 539 metric tons of CO_2 . (Meta offset 100% of these emissions via its sustainability program.) The thesis uses this source for authoritative data on LLaMA 2's carbon footprint, and notes Meta's claim that releasing LLaMA openly will avoid others repeating those training emissions.
6. Meta AI (2024). "Llama 3.1 Model Card and Technical Report." – Meta's documentation for the LLaMA 3 series (which includes a 405B-parameter model) provides energy and emissions metrics for training. Training LLaMA 3 models consumed a cumulative 39.3 million GPU-hours on NVIDIA H100 systems, with an estimated 11,390 tons of CO_2 emitted (location-based) . Thanks to 100% renewable energy matching, Meta reports net zero market-based emissions . This reference was used to compare the scale of LLaMA 3's carbon footprint against OpenAI's models; it highlights how model size scaling (GPT-4 and LLaMA 3) dramatically increases energy demands, and how Meta mitigates this through offsets.
7. Stanford Institute for Human-Centered AI (2023). AI Index 2023 Annual Report. Stanford University. – An authoritative compilation of AI trends, this report includes data on energy usage and CO_2 emissions for major LLMs. It cites research (Luccioni 2022) showing GPT-3's training released 502 tCO_2, roughly 20× more than the 25 tCO_2 for BLOOM . The index situates such comparisons in a broader context (model size, data center efficiency, power source), supporting the thesis section with independent, institutionally vetted statistics on ChatGPT-vs-LLaMA environmental impact.
8. Bender, Emily M., et al. (2021). "On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?" Proc. of ACM FAccT 2021. – This peer-reviewed paper critiques the unchecked scaling of large language models, including an environmental perspective. The authors highlight the opaque yet enormous carbon footprint of training ever-larger models, referencing GPT-3 as a case in point, and call for transparency about energy use . This source was used in the thesis to underscore the ethical imperative of reporting and comparing carbon footprints of models like GPT-4 and LLaMA, reinforcing why efficiency and sustainability must be part of the discussion.

2.4 Key Referecence

1. Fedus, W., Zoph, B., & Shazeer, N. (2022). Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. Journal of Machine Learning Research, 23(120), 1–39. This paper introduces a sparsely-activated large language model (a Mixture-of-Experts Transformer) that achieves up to a 7× faster pre-training with the same computational resources by activating only a subset of model parameters per input . It demonstrates how model sparsity can significantly improve training efficiency and reduce energy consumption, enabling trillion-parameter models to be trained with substantially less computational cost .
2. Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., & Chen, W. (2022). LoRA: Low-Rank Adaptation of Large Language Models. In International Conference on Learning Representations (ICLR 2022). Describes a parameter-efficient fine-tuning technique that dramatically reduces the number of trainable parameters (by up to 10,000×) and GPU memory requirement (3×) when

adapting large pre-trained models . By freezing the original model and only learning small low-rank weight updates, LoRA enables reusing LLMs for new tasks with minimal computational overhead and no added inference latency , highlighting an emerging strategy to curb additional training emissions for downstream applications.

3. Muir, D. R., & Sheik, S. (2025). The road to commercial success for neuromorphic technologies. Nature Communications, 16, Article 3586. This perspective discusses neuromorphic computing – brain-inspired hardware (e.g. spiking neural networks) – as a path to ultra-efficient AI. It notes that neuromorphic processors can achieve orders-of-magnitude lower power usage than conventional GPUs on certain tasks, with reported improvements of 4× to 1700× in power efficiency for specialized sensory workloads and several orders of magnitude less energy in real-time inference scenarios . Such radical hardware efficiency gains, though experimental, point to underexplored avenues for reducing the energy and carbon footprint of AI systems.
4. Xu, K., Sun, D., Tian, H., Zhang, J., & Chen, K. (2025). GREEN: Carbon-efficient Resource Scheduling for Machine Learning Clusters. In Proceedings of the 22nd USENIX Symposium on Networked Systems Design and Implementation (NSDI '25) (pp. 999–1014). USENIX Association. Presents a carbon-aware scheduling framework for AI workloads that cuts emissions by aligning computation with greener energy availability. The proposed cluster scheduler (GREEN) dynamically shifts or delays machine learning jobs to times of lower grid carbon intensity, achieving a 41% reduction in cluster-wide CO_2 footprint with only a modest 3–6% increase in job completion time . This work demonstrates how intelligent scheduling and timing of LLM training can substantially lower emissions by exploiting renewable energy cycles.
5. Xiong, Z., Wang, Y., Stewart, A. J., Heidler, K., Wang, Y., … & Zhu, X. (2024). On the Foundations of Earth and Climate Foundation Models. arXiv preprint arXiv:2405.04285. In this article, the authors argue that reusing and adapting foundation models (pre-trained large models) across multiple tasks offers a sustainable alternative to training separate models from scratch. They emphasize that training new models for every use case incurs significant and redundant carbon cost, whereas fine-tuning one pre-trained model for many purposes dramatically cuts computational demand and associated emissions . This highlights model reuse as a promising strategy to reduce the life-cycle carbon footprint of LLM development.
6. Schneider, I., Xu, H., Benecke, S., Patterson, D., Huang, K., Ranganathan, P., & Elsworth, C. (2025). Life-Cycle Emissions of AI Hardware: A Cradle-to-Grave Approach and Generational Trends. arXiv preprint arXiv:2502.01671. This Google technical report provides the first comprehensive life-cycle assessment (LCA) of AI accelerators, accounting for manufacturing and end-of-life emissions in addition to energy use . It introduces a "compute carbon intensity" metric and shows that newer hardware generations (TPU v6e) have about 3× lower carbon intensity (CO_2 per unit of compute) than their predecessor . The findings illustrate how optimizing hardware design and lifespan can significantly reduce the overall carbon footprint of large-scale AI training and inference, beyond just improving runtime efficiency.
7. Schwartz, R., Dodge, J., Smith, N. A., & Etzioni, O. (2020). Green AI. Communications of the ACM, 63(12), 54–63. This influential position paper calls for greater transparency in reporting energy usage and efficiency in AI research. The authors propose treating computational efficiency as a primary evaluation metric alongside accuracy and suggest that researchers report the "price tag" (compute/energy cost) of training and running models in publications . By making power and carbon costs explicit, this approach encourages the community to prioritize methods that achieve similar results with less energy, thereby steering LLM development toward more sustainable and transparent practices .

2.5 Annotated References

1. U.S. Department of Energy – “AI for Energy” (2023). This DOE report outlines how AI—including large language models—can accelerate power grid modeling, support compliance with federal permitting, forecast renewable energy, improve smart-grid resilience and optimize electric-vehicle charging . It provides an authoritative policy context for using LLMs in energy systems.
2. He & colleagues – “LLM Interface for Home Energy Management Systems” (arXiv, 2025). The authors introduce an LLM-driven interface that interprets natural-language inputs and outputs structured parameters for home energy management. Their experiments demonstrate 88 % accuracy in parameter retrieval, lowering barriers to residential demand-response participation and highlighting the potential for LLMs to enable decarbonization through demand-side flexibility .
3. Gao et al. – “Advancing Building Energy Modeling with LLMs” (Energy & Buildings, 2024). This paper explores how ChatGPT can automate building-energy modeling tasks (e.g., simulation input generation, anomaly detection), reducing engineering effort and improving model accuracy . It serves as a proof of concept for integrating LLMs into building-energy workflows.
4. Jiang et al. – “Large Language Models for Building Energy Applications” (Building Simulation, 2025). A perspective article that discusses LLMs for automated energy model generation, energy management optimization and fault detection, while identifying challenges such as high computational demand . It proposes a development roadmap involving domain-specific fine-tuning and multimodal integration.
5. Ma et al. – “LLM-Empowered Interactive Load Forecasting” (arXiv, 2025). The authors propose a multi-agent system where an LLM interacts with specialized forecasting agents and human operators. The framework improves forecasting accuracy and remains cost-effective, demonstrating how human–LLM collaboration can optimize grid operations .
6. Shu et al. – “CarbonChat: Large Language Model-Based Corporate Carbon Emission Analysis” (arXiv, 2025). This system employs retrieval-augmented LLMs to analyze corporate sustainability reports across 14 GHG dimensions, producing accurate carbon reports and climate Q&A at lower cost .
7. Koujan et al. – “A Multi-Agent Framework for Extracting Carbon Reduction Actions” (ClimateChange.AI workshop, 2024). The authors develop a multi-agent architecture that uses an LLM and retrieval modules to extract carbon reduction levers from corporate reports, creating a sector-specific library of mitigation actions .
8. Zhang et al. – “PCF-RWKV: Large Language Model for Product Carbon Footprint Estimation” (Sustainability, 2024). This paper introduces a RWKV-based LLM with low-rank adaptations for automated product carbon footprint assessment. The model uses multi-agent collaboration to build life-cycle inventories and can run on a single consumer GPU, greatly improving efficiency and data security .
9. Ullah et al. – “Local Climate Services for All, Courtesy of LLMs” (Communications Earth & Environment, 2024). The authors build a prototype (“ClimSight”) showing that LLMs can summarize and convey localized climate information to individuals cost-effectively , demonstrating the potential for democratizing climate services .
10. AppTek et al. – “ClimateGPT” (Blog, 2023). A domain-specific LLM trained on 300B climate-related tokens, providing multilingual climate information and policy support . Although a corporate source, it highlights industry efforts to create specialized LLMs for climate intelligence.
11. Arslan et al. – “Driving Sustainable Energy Transitions with a Multi-Source RAG-LLM System” (Energy & Buildings, 2024). This paper (as summarized by the authors’ repository) introduces an

Energy Chatbot that uses multi-source retrieval-augmented generation to support SMEs in navigating sustainable energy initiatives , offering decision support and reducing costs.

12. UNESCO & UCL – “Smarter, Smaller, Stronger: Resource-Efficient AI and the Future of Digital Transformation” (2025). The report advocates for pivoting away from large, general-purpose models towards smaller, task-specific ones; using mixture-of-experts architectures; writing concise prompts; and applying model compression. Their experiments show these changes can reduce energy use by up to 90 % without sacrificing accuracy , emphasizing design choices that make LLMs themselves greener.
13. Ding et al. – “Tracking the Carbon Footprint of Global Generative Artificial Intelligence” (Innovation, 2025). This large-scale study compiles data on 369 generative AI models and finds that most are deployed in regions with high carbon intensity. They estimate total consumption of 24.97–41.104 TWh and emissions of 10.67–18.61 Mt CO_2 between 2018–2024, suggesting that relocating training and inference to low-carbon regions could significantly reduce emissions .