Green Gradients: A Loss-Landscape-Aware Hypergradient Meta-Optimizer for Eco-Friendly Transformer Training

Ashvik Nacham [1]

Abstract

Training large transformer models incurs substantial computational costs, with a significant portion arising from hyperparameter search. This research introduces **Green Gradients**, a dual-phase, calculus-based meta-optimizer designed to reduce the carbon footprint of transformer fine-tuning. Phase 1 replaces traditional expensive Bayesian learning rate search with a single exponential scan of the loss landscape, where $\partial\mathcal{L}/\partial\lambda$ is estimated to evaluate candidates and identify an optimal initial learning rate λ^* based on the decrease in loss. Phase 2 adapts λ throughout training via hypergradient descent by computing $\partial\mathcal{L}/\partial\lambda$ through the chain rule. Evaluated on DistilBERT applied to the CLIMATE-FEVER dataset across 25 paired trials, Green Gradients achieves a **48.87% reduction** in both energy consumption and carbon emissions compared to an Optuna + REDUCELRONPLATEAU baseline ($p = 1.51 \times 10^{-11}$) with no statistically significant change in test accuracy ($p = 0.106$), demonstrating the viability of Green Gradients in reducing the carbon footprint of transformer training while retaining accuracy.

1. Introduction

The environmental cost of training large language models (LLMs) has grown substantially as model scale increases (Schwartz et al., 2020). A frequently overlooked contributor to this is *hyperparameter search*: Bayesian optimization methods such as Optuna (Akiba et al., 2019) require repeated full training runs to estimate the efficacy of each candidate learning rate. Reactive schedulers such as REDUCELRONPLATEAU wait for training to stall before adjusting λ and cannot *increase* step size when needed.

We propose **Green Gradients**, which addresses both limitations. By profiling the loss landscape mathematically in a single pass and then adapting λ via hypergradient descent, the optimizer eliminates multi-trial search overhead while remaining responsive to the model's evolving gradient dynamics.

2. Related Work

Hyperparameter optimization. Bayesian optimization (Akiba et al., 2019) and random search (Bergstra & Bengio, 2012) require repeated full training runs to evaluate each candidate configuration. Green Gradients replaces this iterative paradigm with a single gradient-informed loss-landscape scan.

Learning-rate schedulers. Cyclical rates (Smith, 2017) and warm restarts (Loshchilov & Hutter, 2017) adapt λ on a fixed schedule. REDUCELRONPLATEAU conditions reductions on validation-loss plateaus but cannot raise λ and reacts only after stagnation. Hypergradient descent (Baydin et al., 2018) adapts λ online using gradient information. Green Gradients' contribution is the specific *combination* of (i) a single-pass, finite-difference loss-landscape scan that replaces multi-trial search for initialization, with (ii) a smoothed, sign-based *log-space* hypergradient update that bounds the relative step independently of gradient scale—applied end-to-end with the explicit goal of minimizing hyperparameter-search energy. This pairing—gradient-informed initialization plus scale-robust online adaptation—distinguishes Green Gradients from prior additive hypergradient methods (Baydin et al., 2018) and applies it to transformer fine-tuning.

Green AI. Schwartz et al. (2020) highlight reporting of computational costs as a first step toward sustainable AI. Our work acts on this goal by optimizing the training pipeline itself.

3. Methodology

Green Gradients is organized around a single mathematical object, $\partial\mathcal{L}/\partial\lambda$, which Phase 1 estimates discretely at a fixed reference point and Phase 2 computes analytically along the training trajectory.

[1]University High School, Irvine, CA, USA. Correspondence to: Ashvik Nacham <ashviknacham@gmail.com>.

Published as a workshop paper at SCALE – ICML 2026. Copyright 2026 by the author(s).

3.1. Phase 1: Loss Landscape Profiling

Objective. Let $\mathcal{L}(\boldsymbol{\theta})$ be the non-convex empirical loss for parameters $\boldsymbol{\theta} \in \mathbb{R}^d$ and $\lambda \in \mathbb{R}_{>0}$ the scalar learning rate. Phase 1 solves

$$\lambda^* = \arg\max_{\lambda} \left(-\tfrac{\partial \mathcal{L}}{\partial \lambda}\right) \text{ s.t. } \mathcal{L}(\lambda) < \mathcal{L}_{\text{div}}, \tag{1}$$

where $\mathcal{L}_{\text{div}}$ is a divergence threshold; violating candidates are discarded.

Exponential scan. $N = 25$ candidates are spaced log-uniformly over $[\lambda_{\min}, \lambda_{\max}]$:

$$\lambda_i = \lambda_{\min} \cdot e^{ki}, \quad i = 0, \ldots, N-1, \tag{2}$$

with $k = \frac{1}{N-1} \ln(\lambda_{\max}/\lambda_{\min})$.

Discrete gradient estimation. At each λ_i the model is reset to fixed weights $\boldsymbol{\theta}_0$ and forward-passed on a small batch to obtain $\mathcal{L}_i$. The hypergradient is estimated by a first-order finite difference:

$$\left.\frac{\partial \mathcal{L}}{\partial \lambda}\right|_{\lambda_i} \approx \frac{\mathcal{L}_i - \mathcal{L}_{i-1}}{\lambda_i - \lambda_{i-1}}. \tag{3}$$

Resetting to $\boldsymbol{\theta}_0$ before each probe eliminates inter-evaluation bias; candidates with high intra-batch variance in $\mathcal{L}$ are additionally excluded as unstable.

3.2. Discrete-to-Continuous Transition

Equations (3) and (5) estimate the same quantity—sensitivity of loss to λ—by different means. In Phase 1, $\partial\mathcal{L}/\partial\lambda$ is a finite difference over candidates λ_i evaluated at the fixed reference $\boldsymbol{\theta}_0$, yielding the one-time selection $\lambda_0 := \lambda^*$. In Phase 2, because $\boldsymbol{\theta}_{t+1}$ is an explicit differentiable function of λ_t via Equation (4), the same derivative is computed analytically along the moving trajectory $\{\boldsymbol{\theta}_t\}$ at zero additional forward-pass cost: ∇_t and ∇_{t+1} are already available from ordinary backpropagation.

3.3. Phase 2: Hypergradient Descent

Chain-rule derivation. At step t, parameter updates follow

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \lambda_t \, \nabla_t, \tag{4}$$

where $\nabla_t := \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}_t)$. Since $\boldsymbol{\theta}_{t+1}$ depends explicitly on λ_t, $\partial\boldsymbol{\theta}_{t+1}/\partial\lambda_t = -\nabla_t$. The chain rule then gives the *hypergradient*:

$$\frac{\partial \mathcal{L}_{t+1}}{\partial \lambda_t} = \nabla_{t+1} \cdot \frac{\partial \boldsymbol{\theta}_{t+1}}{\partial \lambda_t} = -\nabla_{t+1}^{\top} \nabla_t. \tag{5}$$

A positive inner product signals aligned gradient steps and calls for a larger λ; a negative value signals overshoot. The update is *state-aware*, not schedule-driven.

Log-space update & EMA smoothing. The raw additive rule $\lambda_{t+1}^{\text{raw}} = \lambda_t + \beta(\nabla_{t+1}^{\top} \nabla_t)$ is unstable when gradient magnitudes are large. We instead update in log-space,

$$\lambda_{t+1} = \lambda_t \cdot \exp\Big(\beta_{\log} \, \text{sign}\big(\nabla_{t+1}^{\top} \nabla_t\big)\Big), \tag{6}$$

guaranteeing $\lambda_{t+1} > 0$ and bounding the relative step independently of gradient scale. High-frequency noise is suppressed by an EMA of gradient vectors,

$$\tilde{\nabla}_t = \alpha \, \tilde{\nabla}_{t-1} + (1-\alpha) \, \nabla_t, \quad \alpha \in (0,1), \tag{7}$$

with $\tilde{\nabla}_t$ substituted for ∇_t in Equation (5).

3.4. Comparison to Baseline

Table 1 summarises the design differences across five key axes.

4. Experimental Setup

Model and task. We fine-tune DistilBERT (`distilbert-base-uncased`, ≈66M parameters) on a binary version of CLIMATE-FEVER, retaining only the SUPPORTS (label 0) and REFUTES (label 1) claim–evidence pairs from the HuggingFace test partition (Diggelmann et al., 2020). The resulting 2-class dataset is split 70 / 15 / 15% into train / validation / test sets using a fixed seed, yielding approximately 716 training, 153 validation, and 154 test examples. All inputs are tokenized to a maximum of 128 tokens.

Baseline. Optuna (Bayesian optimization) selects the initial λ; REDUCELRONPLATEAU adapts it during training.

Measurement. Energy and CO_2-equivalent emissions are tracked via CodeCarbon (Lottick et al., 2019). Accuracy is reported on a held-out test split.

Implementation details. Phase 1 scans $N = 25$ candidates over $[\lambda_{\min}, \lambda_{\max}] = [10^{-7}, 10^{-3}]$ from fixed weights $\boldsymbol{\theta}_0$; divergence is flagged when $\mathcal{L}_i > 2\mathcal{L}_0$. Phase 2 uses $\beta_{\log} = 0.05$ and $\alpha = 0.9$. Both methods use the AdamW optimizer (weight decay 0.01, gradient-clip norm 1.0) and train for up to 20 epochs with early stopping (patience = 6 evaluation steps, evaluated every 100 gradient steps). Batch size is 16. The Optuna baseline runs 5 trials per seed; REDUCELRONPLATEAU uses factor 0.5, patience 3. All experiments use a Google Colab NVIDIA T4 GPU.

Statistical design. Twenty-five paired trials share random seeds, isolating optimizer choice as the sole variable. Paired two-sided t-tests assess significance.

Table 1. Green Gradients vs. Optuna + REDUCELRONPLATEAU.

Dimension	Baseline	Green Gradients
Hyperparameter selection	Bayesian search (multi-trial)	Loss landscape profiling (single scan)
In-training adaptation	Time-aware, decay-only scheduler	State-aware hypergradient descent
Gradient signal	None (schedule-driven)	$\partial\mathcal{L}/\partial\lambda$ via chain rule
Stability mechanism	Fixed multiplicative decay	Log-space updates + EMA
Energy overhead	High — repeated full training runs	Low — 25-point scan + adaptive updates

5. Results

Energy and emissions. Green Gradients reduced mean total energy from 0.005137 to 0.002525 kWh and mean CO_2e from 0.713 to 0.350 g—a **48.87% reduction** on both metrics ($p = 1.51 \times 10^{-11}$). The reduction was consistent across all seeds (range 30–70%; Appendix C, Figure 4). As shown in Appendix A (Figure 2), this reduction is driven almost entirely by eliminated search overhead: Optuna's Bayesian search phase ($\approx$0.0035 kWh) alone consumes more than twice the energy of the full Green Gradients pipeline (scan + training, $\approx$0.0025 kWh), while per-run training energy is comparable across methods. This rules out under-training as an explanation.

Accuracy. Mean test accuracy was 72.41% (baseline) vs. 70.66% (Green Gradients); the difference is not statistically significant ($p = 0.106$, 95% CI $[-3.91, +0.40]$ pp). Efficiency is the primary benefit, not an accuracy trade-off.

Pareto separation. Figure 1 shows accuracy vs. total energy across all 25 seeds. Green Gradients (orange) clusters below 0.003 kWh while Optuna (blue) ranges from 0.0037 to 0.0077 kWh with no overlap, confirming a clean efficiency separation. The paired energy scatter (Appendix B, Figure 3) shows every individual Green trial below the equality line—i.e., Green used less energy than Optuna on every single seed. Per-trial values appear in Appendix D.

6. Discussion

Bidirectional adaptation. Unlike REDUCELRONPLATEAU, which can only decrease λ, Equation (5) occasionally *increases* λ mid-training when $\nabla_{t+1}^{\top}\nabla_t$ is strongly positive—indicating that a larger step would accelerate convergence. This bidirectionality is a direct consequence of the state-aware signal.

Accuracy variability. Although the mean accuracy gap of 1.75 pp is not statistically significant, Green Gradients trails slightly on average, with occasional seeds where the single-scan initialization underperforms multi-trial search. The per-seed accuracy gain (Green − Optuna) is centred near zero but exhibits a left tail, motivating the warm-restart and multi-hyperparameter extensions noted below.

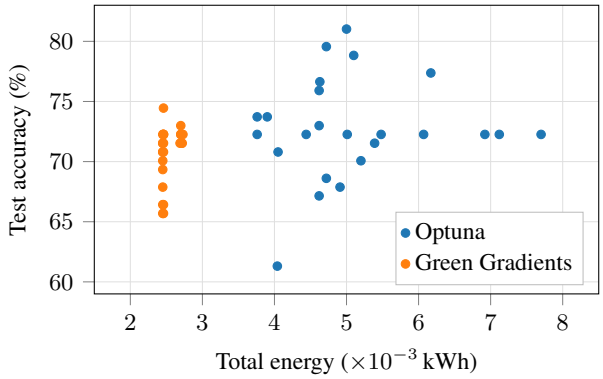

Figure 1. Accuracy vs. total energy (25 paired seeds). Green Gradients (orange) clusters below 3×10^{-3} kWh with comparable accuracy; no Optuna trial (blue) reaches the energy level of any Green Gradients trial.

Initialization sensitivity. In one representative trial, Phase 1 selected $\lambda^* = 4.38 \times 10^{-5}$, a value $1.79\times$ larger than Optuna's 2.45×10^{-5}, supporting faster early convergence; on rougher loss landscapes, Equation (1) selected a more conservative λ^*.

Fair comparison. A potential concern is that comparing Green Gradients' single scan to Optuna's multi-trial method is unfair. However, both methods perform the same task of selecting a learning rate and training the model. Appendix A (Figure 2) shows that Optuna's search phase alone exceeds the entire Green Gradients pipeline energy cost. The efficiency gain is achieved by replacing probabilistic search with a gradient-based scan.

Scope of the environmental claim. The absolute per-run emissions measured here are modest (sub-gram CO_2e per run), reflecting the scale of a single DistilBERT fine-tuning job on a T4 GPU. Our result is therefore best understood as a *relative* efficiency gain rather than a large absolute saving. The benefit scales with the cost of the search being replaced: eliminating multi-trial Bayesian search over larger models and longer jobs would correspond to substantially larger absolute savings; we do not extrapolate such a figure from this experiment alone.

Limitations. Our evaluation covers a single model–task pair (DistilBERT on CLIMATE-FEVER) on one GPU class (NVIDIA T4) and optimizes only λ. The finite-difference estimator (Equation (3)) is sensitive to batch-level stochasticity, and Phase 1 results may further depend on the chosen scan range $[\lambda_{\min}, \lambda_{\max}]$, number of candidates N, and divergence threshold $\mathcal{L}_{\text{div}}$; sensitivity analysis across these parameters is left to future work. The sign-based log-space update (Equation (6)) could slow adaptation in very flat loss regions. Comparison against cheaper scheduling baselines—such as a fixed learning rate, cosine decay, OneCycleLR, and warmup with linear decay—as well as fewer-trial random search variants, is also deferred to future work; Optuna with five trials was selected as the primary baseline because it represents the most common automated hyperparameter-optimisation practice for transformer fine-tuning. Formal convergence guarantees and stability analysis in the non-convex setting are beyond the scope of this empirical study and remain an open theoretical direction. Larger-scale validation and multi-hyperparameter extensions are left to future work.

7. Conclusion

Green Gradients achieves a 48.87% reduction in energy and carbon emissions for transformer fine-tuning with no statistically significant accuracy penalty by replacing probabilistic search with a calculus-based approach: a single exponential scan initializes λ and hypergradient descent with log-space updates and EMA smoothing adapts it dynamically. Future work will evaluate larger architectures, extend the framework to joint multi-hyperparameter optimization, and benchmark against a broader set of schedulers.

Acknowledgements

The author thanks Professor Ioannis Panageas for reviewing the manuscript and providing valuable feedback.

LLM Usage Disclosure

The authors used a large language model assistant for language polishing and LaTeX formatting of this manuscript. All technical content, experimental design, implementation, results, and analysis are the authors' own work.

References

Diggelmann, T., Boyd-Graber, J., Bulian, J., Ciaramita, M., & Leippold, M. (2020). CLIMATE-FEVER: A dataset for verification of real-world climate claims. *NeurIPS Workshop on Tackling Climate Change with ML*.

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. *Proceedings of KDD*, 2623–2631.

Baydin, A. G., Cornish, R., Rubio, D. M., Schmidt, M., & Wood, F. (2018). Online learning rate adaptation with hypergradient descent. *ICLR*.

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *JMLR*, 13, 281–305.

Loshchilov, I., & Hutter, F. (2017). SGDR: Stochastic gradient descent with warm restarts. *ICLR*.

Lottick, K., et al. (2019). Energy usage reports: Environmental awareness as part of algorithmic accountability. *NeurIPS Workshop on Tackling Climate Change with ML*.

Schwartz, R., Dodge, J., Smith, N. A., & Etzioni, O. (2020). Green AI. *Communications of the ACM*, 63(12), 54–63.

Smith, L. N. (2017). Cyclical learning rates for training neural networks. *WACV*, 464–472.

A. Energy Decomposition

Figure 2 decomposes mean per-trial energy by phase. Optuna's Bayesian search ($\approx$0.0035 kWh) accounts for more than the entire Green Gradients pipeline combined. Training energy is nearly identical between the two methods (0.00165 vs. 0.00170 kWh), confirming that the 48.87% total reduction comes exclusively from eliminated search overhead, not from any difference in training effort.

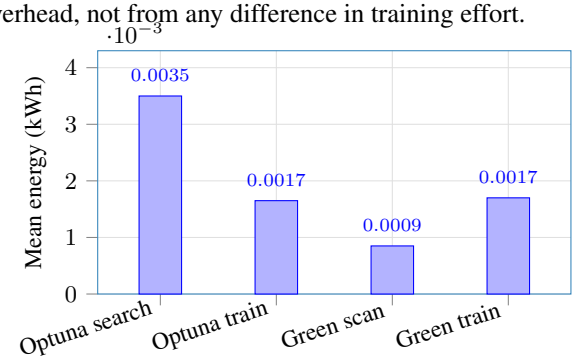

Figure 2. Mean energy by phase. Optuna's search phase dominates its budget. Green Gradients' scan costs less than one-quarter of Optuna's search, while training energy is comparable.

B. Paired Energy Scatter

Figure 3 plots Green Gradients energy against Optuna energy for each of the 25 paired seeds. All 25 points fall below the $y = x$ equality line, confirming that Green Gradients used less energy than Optuna on *every individual seed*—not merely on average.

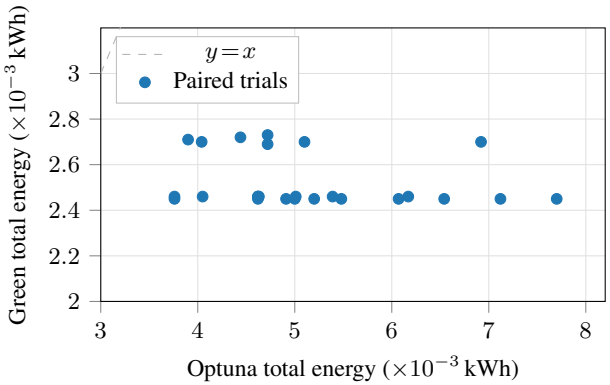

Figure 3. Paired energy per seed (Green vs. Optuna). All 25 points lie below the $y = x$ line: Green Gradients used less energy than Optuna on every individual seed, not just on average.

C. Energy Reduction Distribution

Figure 4 shows the distribution of per-seed energy reduction percentages. All 25 seeds yield a positive reduction (range 30–70%), confirming that the 48.87% mean is not driven by outliers and that Green Gradients consistently outperforms the baseline regardless of random seed.

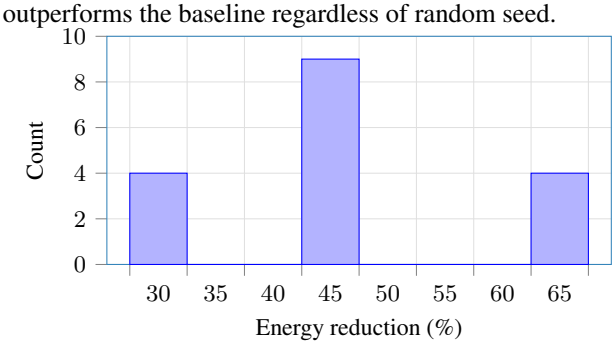

Figure 4. Distribution of per-seed energy reduction (%) across all 25 trials. Every seed yields a positive reduction (range 30–70%), confirming the mean 48.87% result is robust and not driven by outliers.

D. Per-Trial Results

Table 2 reports per-trial accuracy and carbon emissions for all 25 paired seeds.

Table 2. Per-trial accuracy (%) and CO_2e emissions (g).

	Accuracy (%)		**Emissions (g)**	
Trial	Optuna	Green	Optuna	Green
1	72.26	72.26	1.067	0.340
2	72.26	69.34	0.988	0.340
3	77.37	74.45	0.855	0.341
4	72.26	71.53	0.521	0.340
5	71.53	72.26	0.747	0.341
6	64.96	71.53	0.907	0.340
7	81.02	65.69	0.693	0.340
8	73.72	71.53	0.521	0.341
9	70.80	72.26	0.562	0.341
10	72.26	70.80	0.760	0.340
11	75.91	72.26	0.641	0.341
12	61.31	72.26	0.561	0.375
13	78.83	72.99	0.707	0.375
14	72.26	70.80	0.695	0.341
15	73.72	71.53	0.541	0.376
16	72.26	70.07	0.842	0.340
17	79.56	72.26	0.654	0.379
18	67.15	70.80	0.641	0.340
19	70.07	67.88	0.721	0.340
20	72.26	71.53	0.616	0.377
21	72.26	72.26	0.960	0.374
22	76.64	65.69	0.642	0.341
23	67.88	66.42	0.681	0.340
24	68.61	71.53	0.655	0.373
25	72.99	66.42	0.641	0.341
Mean	**72.41**	**70.66**	**0.713**	**0.350**