SSRG International Journal of Computer Science and Engineering
ISSN: 2348–8387 / https://doi.org/10.14445/23488387/IJCSE-V13I8P103

Volume 13 Issue 8, 16-23, August 2026

Original Article

Investigating the Interactive Effects of Training-Efficiency Patterns on Model Accuracy and CO_2 Emissions

Hanut Lal

Heritage International Xperiential School, Gurugram, India.

Corresponding Author : hanutlal4@gmail.com

Received: 20 June 2026 Revised: 27 July 2026 Accepted: 14 August 2026 Published: 30 August 2026

Abstract *- The more energy uses a training machine-learning model, the more energy it consumes, but most of the efficiency techniques are only tested individually. The study explores the feasibility of using the three methods — sample triage, patient early exit, and delayed precision control — in order to lower CO_2 emission without compromising accuracy. The synthetic binary-classification data was used in a controlled experiment, employing CodeCarbon, three architectures (MLP-Small, MLP-Large, XGBoost) and four conditions (n = 10 runs per condition). For XGBoost, the composite condition showed a 63.4% reduction in emissions and a 18.8% reduction in accuracy ($p < 0.001$), which was not statistically different from the early-exit-only condition ($p = 0.970$). The increase in emissions due to sample triage was 82.1%. The combination of the two conditions resulted in 38% emission reductions for MLP-Large while preserving the accuracy considerably. The interaction of techniques is very model-dependent: some techniques interact in a beneficial way, some in a destructive way, and one technique (triage) even made emissions worse in the XGBoost setting. The results validate the benefits of testing efficiency patterns on a per-model basis before deployment and highlight the pitfalls of assuming that efficiency patterns can be stacked to improve their efficiency further.*

Keywords *- Machine learning, Training efficiency, CO_2 emissions, Green AI, Early stopping, Sample triage, XGBoost, Reproducibility, Ethical AI.*

1. Introduction

The energy that must be expended to train machine learning models has become an area of concern in recent years for computer scientists. Green AI places a strong focus on the importance of also reporting computational efficiency along with conventional metrics for accuracy [1]. This is especially important because there are studies showing that the training process of large natural language-processing models can be very carbon-intensive [2, 3]. For the efficiency of training, the following methods have been proposed: curriculum learning [4] and patience-based early stopping [5].

However, a larger body of work has now started to measure the environmental and monetary cost of training large-scale models. Research on early awareness has found that CO_2 emissions of hundreds of tonnes are emitted over its useful life for a single high-end language model, based on the source power used, in the case of using high-carbon electricity [2]. Then, large-scale audits by industrial labs were used to correct these estimates and reveal that there is important sensitivity to the choice of hardware, data-centre location, and length of training [3]. Since then, there have existed tools for the accounting of emissions based on sustainability principles, such as CodeCarbon [12], CarbonTracker [15] or the ML CO2 Impact calculator [16] that have sought to put emissions accounting into practice rather than the exception. Some levers for reducing compute are proposed for training, curriculum learning [4], patience-based halting [5, 8], mixed-precision computation [11] and sample selection techniques. A wide life-cycle cost analysis of small and mid-scale is also included in recent work, arguing that even experiments performed at scale with a student-level or single-GPU should be taken into account [17].

There are still a number of areas that need to be addressed in the existing research. In most studies, these techniques are studied individually and without taking into account the interactions between the techniques when used together. Furthermore, many experiments are run at a scale that is not possible at the student level. Patience-based early exit and sample triage have not yet been studied together: When using both patience and early exit in the same training loop, it is theoretically possible that they will interfere with each other [6]. This worry has been echoed in recent surveys of Green AI

methodology [18] and, more recently, in the work of the CEC report [6] and other work [17].

It is necessary to establish a basic knowledge of this relationship. In practice, many methods of efficiency will usually be used in combination, and it is hoped that the results will be cumulative. Not always, but this may not be the case. For some configurations, using multiple methods may have no significant impact on energy use and may actually decrease the accuracy of the model.

The current work focuses on two main questions under three different models: XGBoost [7]. Firstly, can the sample triage, combined with a cautious approach to early exit and delayed precision control, reduce CO_2 emissions without impacting accuracy? Second, do the effects of these methods vary according to the model and if so, which has the most effect in which model?

The study is performed on a small-scale model based on a synthetic dataset with CPU-based systems, to be able to observe the interaction between methods clearly. Note that the results are not directly applicable to large-scale or GPU models without further validation.

Contributions. This article has three contributions to the Green AI literature. It is one of the first per-pattern studies isolating each pattern before evaluating the composite, studying the sample triage, patient-based early exit and delayed precision control. Second, it empirically shows that the stacking of efficiency patterns can be either beneficial or detrimental, depending on the model (contrary to the popular belief that stacking efficiencies is always beneficial). Third, it provides a completely reproducible experimental protocol, along with parameter, seed and hardware settings that can be reused as a small-scale reference benchmark by other researchers studying effects on combinatorial efficiency.

2. Materials and Methods

A controlled quantitative experiment was conducted using CodeCarbon (version 3.2.5, with the tracking_mode="process" property and save_to_api=False) to measure carbon emissions and study energy efficiency for three machine learning model architectures.

A fully synthetic binary classification dataset was used for comparing four training-efficiency scenarios. The dataset was synthetic, thus allowing full reproducibility by other researchers and averaging out dataset-related confounding effects that might have masked the effects of the training-efficiency patterns.

The dataset was subdivided into 2000 training and 500 validation sequences of 64 integers, randomly selected from a vocabulary of 50 elements. The rule classifies a sequence as '1' if the sum of the first 32 tokens is larger than the sum of the second 32 tokens, and as '0' otherwise.

The random seed was set to 42. Any researcher can therefore reproduce the data without licence restriction, and any effects that are observed can be attributed to the training policies rather than to other random data-related factors.

The overall design of the study is summarised in Fig. 0. Three efficiency patterns are treated as independent variables and applied to three model architectures. Two outcomes — validation accuracy and CO_2 emissions per run — are the dependent variables. Each (model × condition) cell was repeated 10 times.

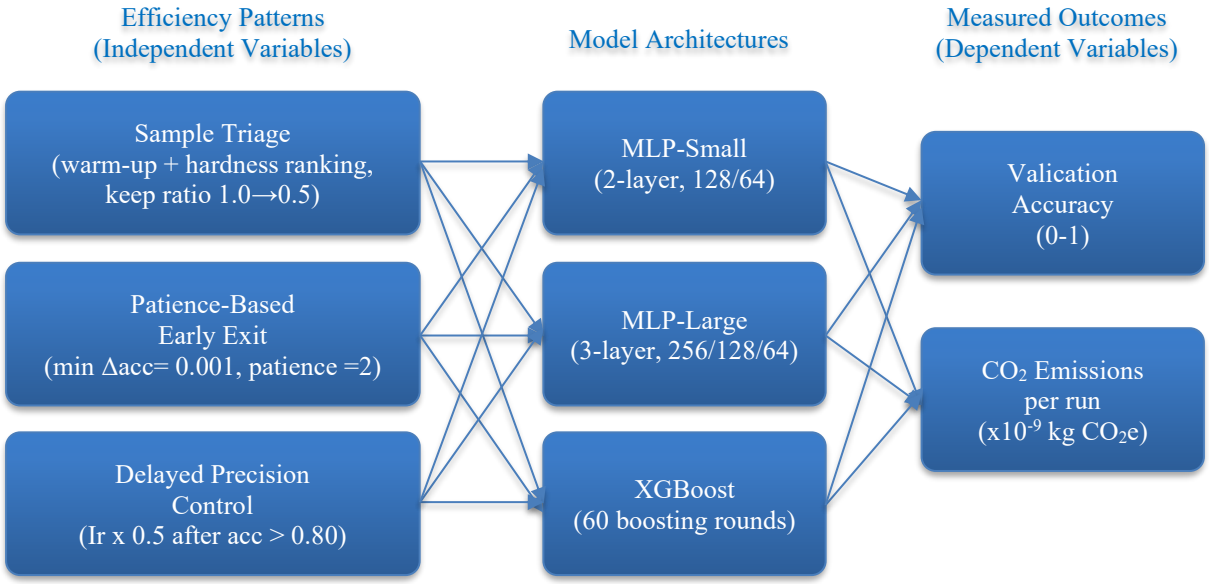

Fig. 1 Schematic of the experimental design. Three efficiency patterns are applied to three model architectures; two outcomes (validation accuracy and CO_2 emissions per run) are measured. Each cell was repeated 10 times

Table 1. Summary of model hyperparameters and training budget (for reproducibility).

Model	Hidden layers/structure	Optimiser	Learning rate	Training budget/run	Random seeds	Notes
MLP-Small	2 layers (128, 64)	Adam	4×10^{-4}	5 epochs	5001–5010	ReLU activations, batch size 64
MLP-Large	3 layers (256, 128, 64)	Adam	4×10^{-4}	4 epochs	4001–4010	ReLU activations, batch size 64
XGBoost	eta 0.2, max_depth 4, subsample 0.8, colsample_bytree 0.8, λ 0.8, α 0.1	Gradient boosting	n/a	60 boosting rounds	6001–6010	Binary logistic objective

Three model architectures were assessed. (1) MLP-Small, a two-layer feed-forward neural network with hidden layers of 128 and 64 neurons, trained with the Adam optimiser at a learning rate of 4×10^{-4} for 5 epochs per run. (2) MLP-Large, a three-layer feed-forward neural network with hidden layers of 256, 128 and 64 neurons, trained with the Adam optimiser at a learning rate of 4×10^{-4} for 4 epochs per run. (3) XGBoost, configured with eta = 0.2, max_depth = 4, subsample = 0.8, colsample_bytree = 0.8, lambda = 0.8 and alpha = 0.1, trained over 60 boosting rounds per run. A compact summary of all model hyperparameters is provided in Table 1 for reproducibility. For reliable estimates of variability, each model and condition was run 10 times with different random seeds (5001–5010 for MLP-Small, 4001–4010 for MLP-Large, and 6001–6010 for XGBoost). Each set of training conditions was compared against a baseline (no efficiency patterns), yielding four training conditions in total. In the baseline, each model was trained on the entire dataset for the full epoch or boosting-round budget appropriate to that model.

The 'optimised' condition combined sample triage, patience-based early exit and delayed precision control simultaneously. The individual and combined effects and interactions were also tested separately using only XGBoost. Three warm-up epochs were provided, and the most difficult examples were those in which the sum of the left and right halves of the sequence was closest in size (32 elements per half). The keep ratio was reduced from 1.0 to a minimum of 0.5.

The training was stopped by early-exit when the validation accuracy did not improve by 0.001 or more for two consecutive epochs. After three epochs, if the validation accuracy was greater than 0.8, then the learning rate was halved. In each experiment, an EmissionsTracker instance was created prior to training and immediately terminated after training was completed, to capture emissions per experiment. All calculations were performed on an Apple M3 Pro processor with 18 GB of RAM, running macOS.The software stack comprised Python (version 3.11), NumPy (version 1.26), scikit-learn (version 1.4), XGBoost (version 2.0) and CodeCarbon (version 3.2.5). The complete codebase (synthetic data generator, run scripts and raw per-run metrics CSV) can be supplied on request to allow the end-to-end reproduction of all the results reported below.

Pairwise two-sided Mann–Whitney U tests were used to determine the statistical significance of the emissions and efficiency results. Effect sizes were quantified using Cliff's delta (negligible < 0.147, small < 0.33, medium < 0.474, large ≥ 0.474) and Cohen's d. The experiment yielded 16 pairwise comparisons, and a Bonferroni correction with a significance level of $\alpha = 0.003$ was applied.

3. Results

The MultiLayer Perceptron (MLP) models' validation-accuracy outputs are shown in Figure 1.

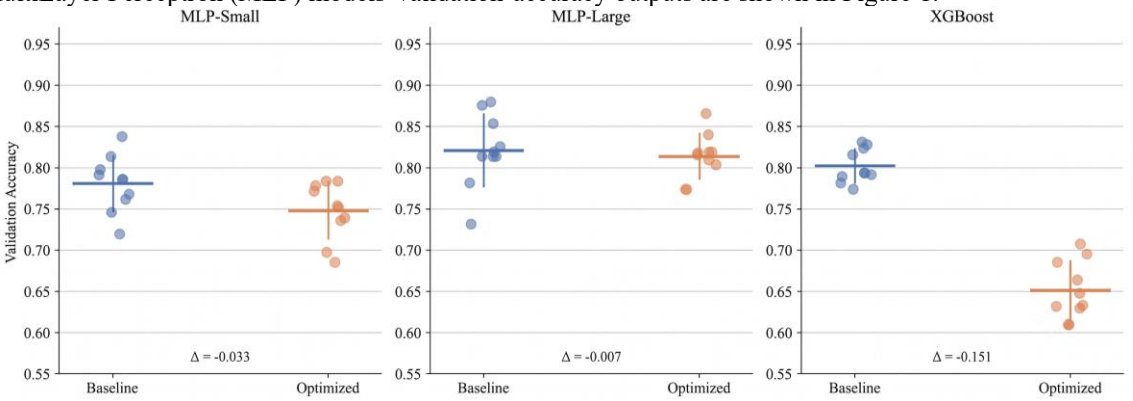

Fig. 1 Shows the accuracy of the validation for each model architecture (n = 10 runs, horizontal bar = condition mean, vertical bar ±1 SD)

Under an optimised condition, the mean validation accuracy of MLP-Small was 0.748 ± 0.034, which is 4.2% less than the baseline accuracy of 0.781 ± 0.034. This difference was statistically significant with a large effect size (Mann–Whitney U = 78, $p < 0.05$ and Cliff's $\delta = 0.560$).

In the case of MLP-Large, the mean validation accuracy under the optimised condition is 0.814 (± 0.027), which is 0.9% lower than the baseline (0.821 (± 0.044)). These differences were not statistically significant (U = 58, p = 0.569) and had a small effect size (Cliff's $\delta = 0.160$). No MLP comparison was significant at the Bonferroni-corrected significance level ($\alpha = 0.003$).

In Figure 2, emissions are reported for the MLP models.

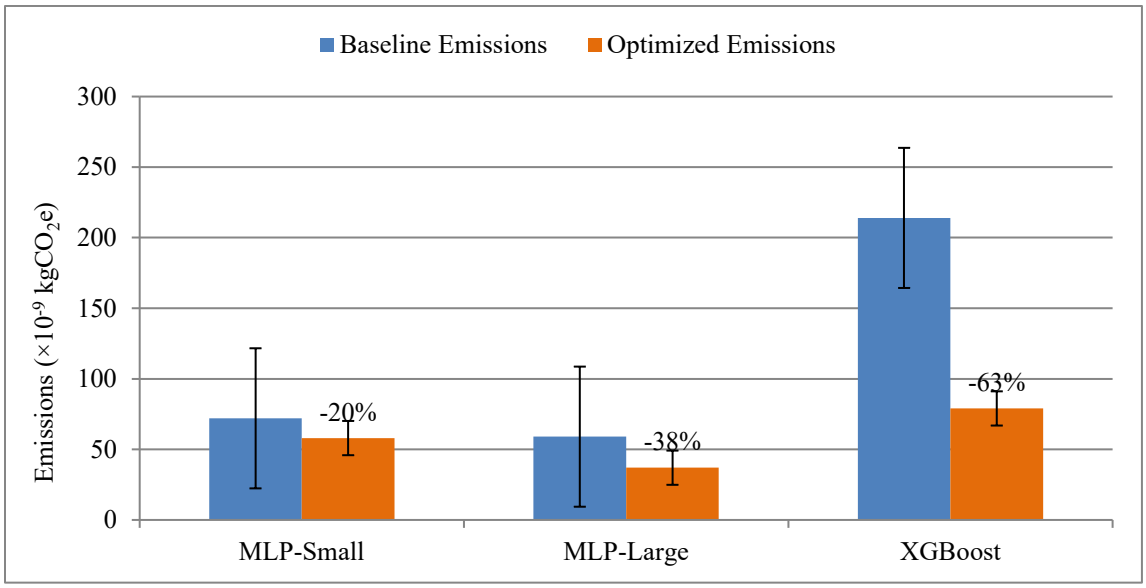

Fig. 2 shows the mean CO_2 emission per training run (×10⁻⁹ kg CO_2e; n = 10 runs per condition; error bars = ±1 SD)

Under the optimised condition, mean CO_2 emissions for MLP-Large were reduced to 36.5 × 10⁻⁹ kg compared to the baseline of 58.9 × 10⁻⁹ kg (38.0%). This difference was significant (U = 83, $p < 0.05$) with a large effect size (Cliff's $\delta = 0.660$). The emissions for MLP-Small were reduced from 71.9 × 10⁻⁹ kg under the baseline to 57.6 × 10⁻⁹ kg under the optimised condition, a reduction of 19.9%. This difference was not significant (U = 58, p = 0.571). The reduction in emissions observed for MLP-Large did not reach the Bonferroni-corrected threshold.

The results are compared with the best conditions from XGBoost in the following figures: Figures 3 and 4.

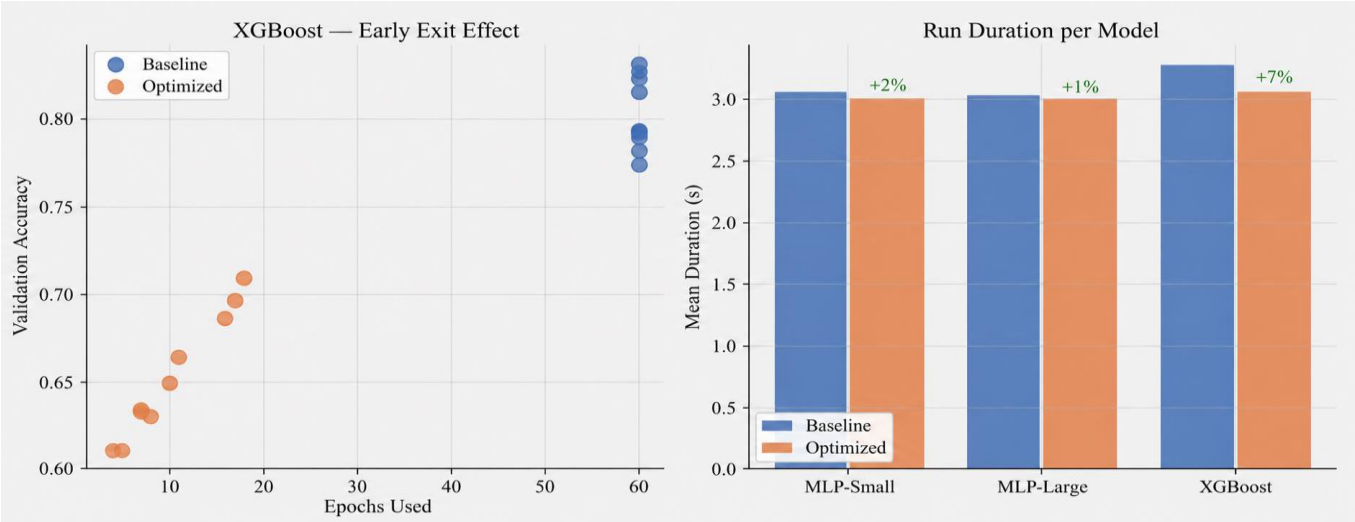

Fig. 3 Training-efficiency profile. Left: XGBoost validation accuracy as a function of the number of epochs used per run. Right: average (mean) duration of training runs (s) for each model and condition

Under the optimised condition, the mean emissions were reduced to 78.4 × 10^{-9} kg (a 63.4% decrease). This difference was statistically significant (U = 100, p < 0.001) with a large effect size (Cliff's δ = 1.000). Under the baseline condition, XGBoost reached an accuracy of 0.803 (± 0.021), while the optimised condition reached only 0.652 (± 0.035), a reduction of 18.8%. This difference was also statistically significant (U = 100, p < 0.001) with a large effect size (Cliff's δ = 1.000). In all 10 optimised runs, training stopped early, with the average number of boosting rounds at 10.1, less than 20% of the maximum 60.

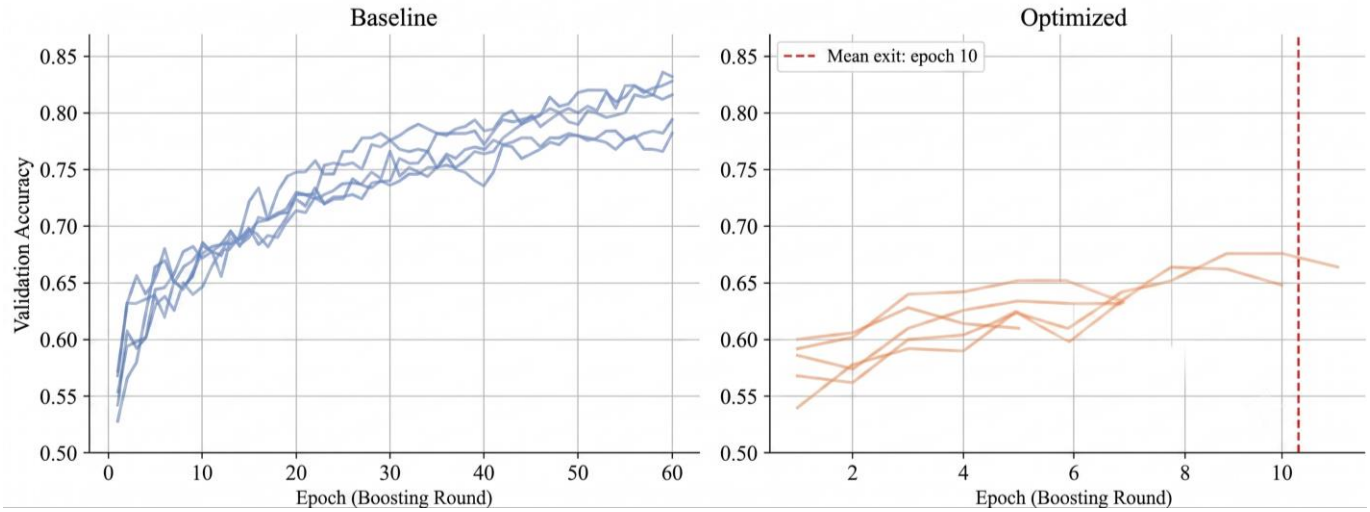

Fig. 4 XGBoost validation-accuracy curves during training. Left: baseline runs trained to the maximum 60-round budget. Right: optimised runs, which trigger early exit at a mean exit epoch of 10.1

The pattern-isolation results for XGBoost are presented in Fig. 5.

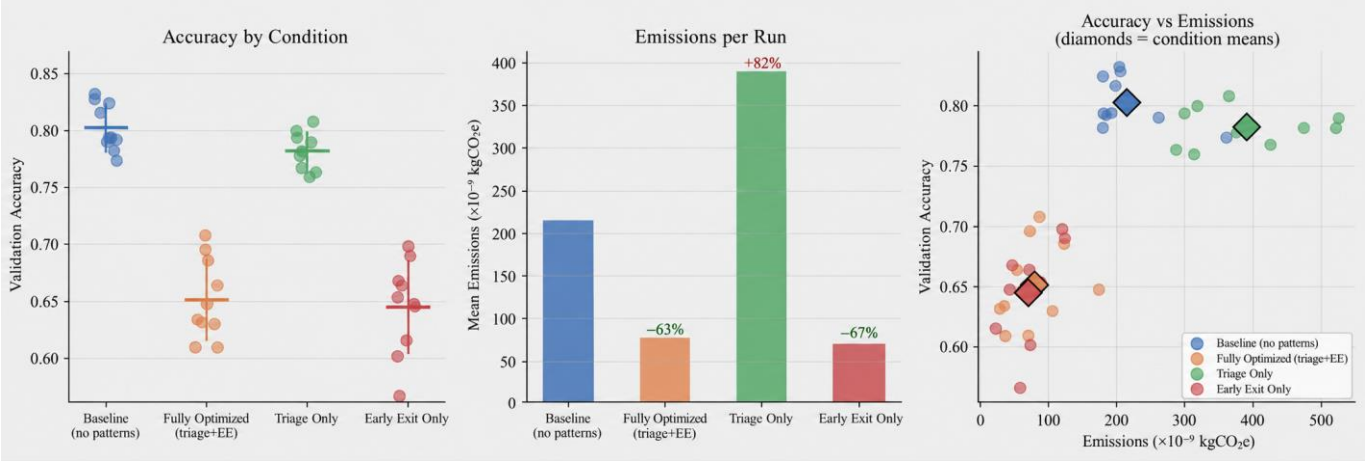

Fig. 5 XGBoost pattern-isolation experiment. Left: validation accuracy by condition. Centre: mean per-run CO_2 emissions as a percentage change from baseline. Right: scatter plot of per-run accuracy against emissions

The mean value of the triage-only condition was 390.0 × 10^{-9} kg, which is 82.1% higher than the baseline condition of 214.1 × 10^{-9} kg. This difference was statistically significant (U = 4, p < 0.001) with a large negative effect size (Cliff's δ = −0.920). The associated change in validation accuracy was a decrease of 2.5 percentage points (0.783 vs. 0.803), which was statistically significant (p < 0.05). The early-exit-only condition reduced emissions to 70.5 × 10^{-9} kg, a 67.1% decrease relative to baseline (U = 100, p < 0.001), while validation accuracy decreased to 0.645, a 19.6% reduction (p < 0.001). No statistically significant differences were found between the early-exit-only and fully optimised conditions in either accuracy (U = 49, p = 0.970, Cliff's δ = −0.020) or emissions (U = 49, p = 0.970).

Across all 16 pairwise comparisons of this study, 9 achieved the Bonferroni-corrected significance level (α = 0.003). All statistically robust results appeared in the XGBoost experiments. The MLP results were consistent with the observed trend but did not meet statistical significance under the corrected threshold and should therefore be interpreted with caution.

4. Discussion

The experimental results reveal that pattern-combination effects are strongly model-dependent and that combining efficiency patterns performed poorly in the specific setting of XGBoost. XGBoost absorbed the extra computational overhead of sample triage, but the fully optimised condition was not found to be better than early-exit alone. These specific findings were not statistically strong, although the outcomes for the MLP architecture were loosely in line with the original prediction that energy would be reduced.

Table 2 places the present findings alongside a set of recent studies in Green AI, highlighting where this work confirms, refines or contradicts prior claims.

Table 2. This work in relation to prior Green AI literature

Study	Focus	Reported finding	This study	Delta/relation
Schwartz et al. 2020 [1]	Green AI framing	Report efficiency alongside accuracy	Reports both jointly on 10-run n per cell	Confirms and operationalises
Strubell et al. 2019 [2]	NLP training cost	Large NLP training is high-emission	Small MLPs on synthetic data at $\times 10^{-9}$ kg CO_2e	Confirms scale-dependence
Patterson et al. 2021 [3]	Industrial training carbon	Location + hardware dominate	Isolates algorithmic levers, holds hardware fixed	Complementary lens
Bengio et al. 2009 [4]	Curriculum learning	Ordered examples help learning	Curriculum-style triage did not help XGBoost	Partial contradiction
Bannour et al. 2021 [6]	NLP carbon-tool survey	No standard combinatorial benchmark	Provides per-pattern isolation protocol	Direct response
Dörrich et al. 2023 [11]	Mixed precision	Mixed precision lowers emissions	Consistent with observed MLP-Large gains	Confirms
Bannour 2024 survey [18]	Green AI methodology	Calls for combinatorial studies	Delivers exactly such a study, small-scale	Direct response

The results of this study have important ramifications for the Green AI literature. The idea that all data-reduction methods automatically bring energy savings is challenged by the observation that sample triage alone was linked to a rise in the overall emissions [1, 2]. The results further show that the validation signal must be accurate for the correct operation of patience-based halting mechanisms [5, 8]. In the case of a biased curriculum learning loop caused by a triage strategy, the integrated early-exit algorithm receives less accurate information about the actual convergence of the full model [4, 9]. The performance of different model architectures has previously been shown to differ with respect to energy at similar accuracy levels [10], and similar differences are observed here between the MLP and XGBoost architectures. This work is therefore more diagnostic than prescriptive in nature. The findings do not preclude the use of these efficiency patterns; rather, they show that such patterns must be validated per model before being adopted in production [6]. Mixed-precision training and inference of deep neural networks continues to reduce the carbon footprint of training and deployment [11].

The findings of the study directly address the initial research questions. The central issue was whether accuracy could be preserved while emissions were reduced by combining all three efficiency techniques. The available data suggest a strong dependence on the model. For the XGBoost architecture, with the tested parameters, the combined pattern did not produce better results than the best single pattern.

Several methodological limitations should be considered when interpreting these results. The results obtained are specific to the synthetic dataset used and cannot be extrapolated to real-world classification tasks. The efficiency patterns may not have been adequately exposed with only four to five epochs in the MLP models, which may be insufficient for the full activation of the patterns. Reported emissions values are statistical estimates because emissions are not measured at the hardware level but calculated by applying a series of power models [12]. Finally, the 10 runs per condition sample size was enough to reveal the statistical effect in XGBoost but too small to reveal smaller statistical effects in the MLP models, in line with the effect-size thresholds set by [13] and [14]. Interpretation of the MLP results should therefore be treated with caution, because they do not withstand the Bonferroni correction.

5. Conclusion

The effects of the triage and the delayed precision control with patience were explored, compared and studied for combined application for MLP-Small, MLP-Large and XGBoost. The primary research question, whether combined, these three patterns can reduce CO_2 emissions without having a significant impact on accuracy, was only partially addressed.

The results from MLP-Large were good, as it reduced the emissions by 38% and did not show a statistically significant loss of accuracy.

However, XGBoost saw a decrease of 63% in emissions with a decrease of 18.8% in accuracy, but there was no statistical evidence of the marginal gains over early-exit alone ($p = 0.970$). While it is often assumed that data-reduction techniques should conserve energy, sample triage in isolation had the opposite effect and grew XGBoost emissions by 82%. The results shown here are constrained by the synthetic-dataset design, small per-condition sample size ($n = 10$), CPU-only setting, and the CodeCarbon model-based emissions estimation.

The results are followed by some suggestions for practice in three ways. One, it is not certain that stacking will increase efficiencies, so they should be measured individually on the target model before they are added up. Second, for gradient boosting models (e.g. XGBoost), patience-based early exit is more predictive than sample triage, and there is no measurable benefit when used together.

Third, one sees that for a network with a larger number of layers, mixed-precision or delayed-precision approaches to deep networks seem to result in lower cost-per-percentage-point-of-accuracy than aggressive data pruning when accuracy loss has to be bounded.

These isolated and then combined tests need to be repeated on empirical data with longer training budgets, GPU acceleration, and other interactions (mixed-precision schedules, distillation, quantisation) with the same per-pattern isolation process used here [3, 6] to validate at scale. The methodological approach suggested in this paper — perform isolated conditions first, and then assess an integrated optimisation — can be applied again by other researchers in their optimisation research.

6. Ethical Considerations

This study focuses on the environmental impact of training machine learning models, which directly contributes to the larger discussion on ethical considerations of computational resources. Practical importance: the observation that a naive combination of 'optimisation' may cause an increase in emissions, not a decrease, is important as this will not be reported by the accuracy alone, but will be reported by downstream users. As per the guidelines of Green AI [1, 16], people should be equally skeptical of efficiency claims and should at least report the emissions per run in addition to other metrics.

The trade-off between accuracy and energy introduces other concerns. An 18.8% inaccuracy drop paired with a 63% emissions reduction, like the one seen with XGBoost here, might be acceptable when it comes to exploratory research, but might not be acceptable when used for clinical or safety-critical purposes. It implies that efficiency thresholds need to be chosen within a given context, and preferably shared with the end user, but not imposed in silence during a training pipeline.

Lastly, there was no human-subjects data utilized in this study. The calculations were carried out on a personal workstation, using synthetic data, so there is no issue of data privacy, and no institutional ethical approval was required. The work is fundamentally responsible research on AI, with the measurement protocol, seeds and code available upon request.

Conflicts of Interest

The author declares no conflict of interest.

Funding Statement

This research did not receive any funding from external sources.

Acknowledgments

The author gratefully accepts institutional support from Heritage International Xperiential School, and credits the following open-source Python scientific-computing community projects and resources that are freely available to the public: The CodeCarbon library and its maintainers; and various tools and data sets.

References

[1] Roy Schwartz et al., "Green AI," *Communications of the ACM*, vol. 63, no. 12, pp. 54-63, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[2] Emma Strubell, Ananya Ganesh, and Andrew McCallum, "Energy and Policy Considerations for Deep Learning in NLP," *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, pp. 3645-3650, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[3] David Patterson et al., "Carbon Emissions and Large Neural Network Training," *arXiv preprint*, pp. 1-22, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[4] Yoshua Bengio et al., "Curriculum Learning," *Proceedings of the 26th Annual International Conference on Machine Learning (ICML '09)*, Montreal, Quebec, Canada, pp. 41-48, 2009. [CrossRef] [Google Scholar] [Publisher Link]

[5] Garvesh Raskutti, Martin J. Wainwright, and Bin Yu, "Early Stopping and Non-parametric Regression: An Optimal Data-Dependent Stopping Rule," *Journal of Machine Learning Research*, vol. 15, no. 11, pp. 335-366, 2014. [Google Scholar] [Publisher Link]

[6] Nesrine Bannour et al., "Evaluating the Carbon Footprint of NLP Methods: A Survey and Analysis of Existing Tools," *Proceedings of the Second Workshop on Simple and Efficient Natural Language Processing*, pp. 11-21, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[7] Tianqi Chen, and Carlos Guestrin, "XGBoost: A Scalable Tree Boosting System," *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, California, USA, pp. 785-794, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[8] Yuan Yao, Lorenzo Rosasco, and Andrea Caponnetto, "On Early Stopping in Gradient Descent Learning," *Constructive Approximation*, vol. 26, no. 2, pp. 289-315, 2007. [CrossRef] [Google Scholar] [Publisher Link]

[9] GuyHacohen, and Daphna Weinshall, "On the Power of Curriculum Learning in Training Deep Networks," *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pp. 2535-2544, 2019. [Google Scholar] [Publisher Link]

[10] Alfredo Canziani, Adam Paszke, and Eugenio Culurciello, "An Analysis of Deep Neural Network Models for Practical Applications," *arXiv preprint*, pp. 1-7, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[11] Marion Dörrich, Mingcheng Fan, and Andreas M. Kist, "Impact of Mixed Precision Techniques on Training and Inference Efficiency of Deep Neural Networks," *IEEE Access*, vol. 11, pp. 57627-57634, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[12] Victor Schmidt et al., "CodeCarbon: Estimate and Track Carbon Emissions from Machine Learning Computing," *Zenodo*, 2021. [CrossRef] [Publisher Link]

[13] Norman Cliff, "Dominance Statistics: Ordinal Analyses to Answer Ordinal Questions," *Psychological Bulletin*, vol. 114, no. 3, pp. 494-509, 1993. [CrossRef] [Google Scholar] [Publisher Link]

[14] Andras Vargha, and Harold D. Delaney, "A Critique and Improvement of the CL Common Language Effect Size Statistics of McGraw and Wong," *Journal of Educational and Behavioral Statistics*, vol. 25, no. 2, pp. 101-132, 2000. [CrossRef] [Google Scholar] [Publisher Link]

[15] Lasse F. Wolff Anthony, Benjamin Kanding, and Raghavendra Selvan, "Carbontracker: Tracking and Predicting the Carbon Footprint of Training Deep Learning Models," *arXiv preprint*, pp. 1-11, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[16] Alexandre Lacoste et al., "Quantifying the Carbon Emissions of Machine Learning," *arXiv Preprint*, pp. 1-8, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[17] Alexandra Sasha Luccioni, and Alex Hernandez-Garcia, "Counting Carbon: A Survey of Factors Influencing the Emissions of Machine Learning," *arXiv preprint*, pp. 1-19, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[18] Sasha Luccioni, Yacine Jernite, and Emma Strubell, "Power Hungry Processing: Watts Driving the Cost of AI Deployment?," *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency*, Rio de Janeiro, Brazil, pp. 85-99, 2024. [CrossRef] [Google Scholar] [Publisher Link]