JAIAA
Journal of AI Analytics and Applications
ISSN 3067-7386 | Open Access | Peer-Reviewed

RESEARCH ARTICLE
Vol. 3, No. 4 (2025), pp. 1-18
DOI: 10.63646/jaiaa.2025.030401

Operationalizing Green AI Analytics: A Reproducible Framework for Monitoring Model Complexity Across Classical and Deep Learning

Peng Zhao[1]; Yifan Li[2]; Min Zhou[3, *]

[1] School of Information, Renmin University of China, Beijing, China
[2] School of Computer Science and Engineering, Southeast University, Nanjing, China
[3] School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing, China
* Corresponding author: m.zhou@bupt.edu.cn

ARTICLE INFO
Received
May 12, 2025
Revised
Augustr 28, 2025
Accepted
October 18, 2025
Available Online
December 20, 2025
DOI
10.63646/jaiaa.2025.030401

Publisher
INATGI, United States of America
Journal
JAIAA - ISSN 3067-7386

Abstract
Green AI has moved the discussion on artificial intelligence from model accuracy alone to the broader question of how computationally sustainable, comparable, and reproducible model development can become. Yet in practice, complexity reporting remains fragmented. Classical machine learning studies often report fit time or prediction speed, deep learning studies usually foreground parameter counts or forward-pass FLOPs, and work on quantized or compressed models frequently emphasizes memory savings without offering a unified account of training-, inference-, and precision-sensitive workload. This article develops a reproducible Green AI analytics framework for operational monitoring of model complexity across both classical and deep learning pipelines. The framework integrates four analytic layers—pipeline metadata, workload metrics, reproducibility controls, and decision analytics—and evaluates 104 standardized benchmark configurations spanning classical learners, compact vision models, encoder models, and large language models under multiple precision regimes. The analysis shows three consistent patterns. First, BOP-aware monitoring changes model rankings in ways that FLOPs-only reporting often misses, particularly under INT8 and INT4 settings. Second, lightweight classical models remain highly competitive on the efficiency frontier for structured-data tasks, whereas compact neural architectures dominate only after accuracy thresholds exceed roughly 0.85. Third, adding reproducibility metadata and phase-specific reporting materially improves cross-study interpretability. Regression results further indicate that precision-aware workload measures explain a larger share of modeled sustainability burden than FLOPs alone. Rather than proposing another isolated profiler, the article demonstrates how Green AI analytics can be operationalized as a reporting and benchmarking practice that supports transparent model selection, deployment governance, and efficiency-aware research design.

Keywords: Green AI; computational workload; reproducibility; FLOPs; Bit-Operations; model complexity; machine learning benchmarking; precision-aware analytics

I. INTRODUCTION

Artificial intelligence research is increasingly shaped by a tension between two desirable goals. On one side lies predictive ambition: larger datasets, broader model families, stronger benchmarks, longer training schedules, and more aggressive search for accuracy. On the other lies sustainability: the need to understand whether performance gains justify the additional computational load, environmental burden, infrastructure cost, and barriers to reproducibility that accompany increasingly heavy models. The recent history of AI

makes this tension difficult to ignore. Performance leaps in language and vision have been driven not only by better architectures but also by substantial increases in compute, data volume, and training complexity, as shown in work on scaling laws, foundation models, and large-scale pretraining (Kaplan et al., 2020; Brown et al., 2020; Bommasani et al., 2021). Once a research community starts accepting compute escalation as normal, model comparison becomes harder, replication becomes costlier, and results may favor groups with the largest infrastructure footprints rather than the strongest ideas.

The Green AI conversation emerged precisely because accuracy-centered reporting was proving insufficient. Research on energy and carbon accounting in machine learning argued that experimental results need to be interpreted together with the resources required to obtain them, especially when model selection depends on repeated tuning and retraining (Henderson et al., 2020; García-Martín et al., 2019; Lannelongue et al., 2021). Parallel work on research reporting emphasized that experimental papers often omit the practical details needed to understand how much computation, hyperparameter search, or engineering effort underlies an apparently simple result (Dodge et al., 2019). At the same time, critical scholarship on language-model scale highlighted that large models create downstream social, ecological, and epistemic costs that are invisible when evaluation is narrowed to benchmark scores alone (Bender et al., 2021). Green AI therefore should not be understood as a moral slogan added after training is complete. It is an analytic program that requires the systematic operationalization of efficiency, comparability, and accountability within the model-development cycle itself.

Despite this conceptual progress, operational practice remains fragmented. In deep learning, model size is often proxied by parameter counts, FLOPs, tokens processed, or latency under a specific hardware stack. In classical machine learning, researchers still tend to report wall-clock training time, fit complexity, or memory use without a consistent framework that can be compared across environments. Quantization and compression studies frequently show dramatic reductions in memory or inference cost, yet their claims may rely on inconsistent metrics: one paper reports compressed weight storage, another reports throughput on a specific accelerator, and a third reports speed-up under a custom runtime. Without a shared monitoring framework, these studies remain difficult to compare. The problem is especially acute when one tries to benchmark classical and deep learning models together. A gradient-boosted tree, a compact convolutional network, and a quantized transformer can solve related decision problems, but the terms in which they describe computational burden are often incompatible.

This article addresses that gap by proposing a reproducible framework for Green AI analytics that treats complexity monitoring as a layered reporting practice rather than as a single number. The framework is motivated by recent methodological work on hardware-independent workload estimation and precision-sensitive accounting, but it is broader in scope. It asks how model complexity should be documented, normalized, interpreted, and linked to deployment decisions when both classical and deep learning pipelines are under consideration. The central argument is that meaningful Green AI analytics must integrate at least four layers: model and pipeline metadata, workload metrics that distinguish training and inference, reproducibility controls that capture environment and experimental provenance, and a decision layer that translates complexity into benchmark rankings, Pareto analysis, and reporting outputs. The article then demonstrates this framework through a controlled benchmark matrix covering multiple model families and precision regimes.

Table I. Taxonomy of Green AI monitoring metrics and their operational roles.

Metric family	Example indicators	Primary analytic use	Main caution

Arithmetic workload	Training FLOPs; inference FLOPs; optimizer/update FLOPs	Architecture comparison and phase separation	May understate quantization benefits when used alone
Precision-aware workload	Training BOPs; inference BOPs; precision divergence ratio	Deployment realism and low-bit monitoring	Depends on effective bit-width assumptions
Storage burden	Parameters; checkpoint size; optimizer state	Model packaging and memory planning	Weak proxy for arithmetic cost
Temporal behavior	Latency; throughput; retraining cadence	Service-level and operational scheduling decisions	Strongly hardware- and runtime-dependent
Provenance & reproducibility	Seeds; versions; logging exports; environment metadata	Cross-study interpretability and re-execution	Often omitted despite high analytical value

Table I distinguishes five complementary metric families. This taxonomy matters because it prevents one indicator from being overloaded with meanings it cannot carry. In the framework proposed here, arithmetic and precision-aware measures anchor the complexity analysis, while storage, temporal, and provenance indicators explain how the same workload profile may lead to very different deployment choices. The table also clarifies why reporting walls of isolated numbers is analytically weak: without an explicit statement of what a metric is intended to explain, comparison tends to become inconsistent.

II. METRIC TAXONOMY AND LITERATURE FOUNDATIONS

Before presenting the benchmark protocol, it is useful to specify the metric taxonomy that underlies Green AI analytics. In practice, workload reporting can be organized into at least five metric families: arithmetic metrics, precision-aware metrics, storage metrics, temporal metrics, and provenance metrics. Arithmetic metrics include forward-pass FLOPs, backward-pass FLOPs, optimizer-update FLOPs, and preprocessing operations. Precision-aware metrics include BOP-style formulations or other bit-scaled proxies that account for the numerical width of operands. Storage metrics include parameter count, checkpoint size, optimizer state, and activation memory. Temporal metrics include latency, throughput, and retraining cadence. Provenance metrics include library versions, seed control, runtime environment, and export trace. A mature analytics framework should not collapse these into one undifferentiated field. Instead, it should specify which family answers which question. Arithmetic metrics are strongest for architecture comparison, precision-aware metrics for deployment realism, storage metrics for serving feasibility, temporal metrics for operational service levels, and provenance metrics for reproducibility.

This taxonomy also clarifies why many current reporting practices frustrate comparison. Parameter count is a storage-oriented metric masquerading as a universal complexity index. Wall-clock training time is a temporal metric heavily conditioned by hardware, implementation quality, and cluster utilization. Memory footprint is meaningful but often incomparable when optimizer state, sequence length, or activation checkpointing differ across runs. In contrast, arithmetic and precision-aware metrics are closer to the intrinsic structure of the pipeline. They do not eliminate all ambiguity, but they make the source of differences easier to interpret. For instance, two models with similar inference FLOPs may differ sharply in BOP burden if one relies on lower-precision arithmetic, and two models with comparable BOP burden may differ sharply in storage if one uses full fine-tuning and the other uses parameter-efficient adaptation. Operational reporting should expose, not hide, these distinctions.

Classical machine learning illustrates the value of this taxonomy particularly well. In many applied studies, fit time for gradient boosting or random forests is reported as though it were self-explanatory. Yet fit time depends on thread count, memory locality, data layout, library implementation, and even whether histogram bins are reused between folds. An arithmetic-oriented Green AI view asks a more transferable question: what kinds of operations dominate the training path, and how do those operations scale with sample size, feature count, and ensemble depth? Once these questions are made explicit, it becomes easier

to explain why an XGBoost model with slightly higher training burden may still be preferable to a neural baseline if inference remains cheap and retraining cycles are infrequent. The same logic applies to logistic regression and linear SVM: their apparent simplicity hides strong advantages in deployment repeatability, maintenance cost, and auditability that deserve a more systematic place in efficiency analysis.

Deep learning, by contrast, often suffers from the opposite problem. Because the community has become comfortable discussing architectures in terms of layers, heads, parameters, and approximate FLOPs, it can appear that complexity is already well understood. In reality, many papers still blur together pretraining cost, fine-tuning cost, inference cost, and adaptation cost. This is especially problematic for foundation-model workflows, where the question faced by most practitioners is not how much the original model cost to train, but how much it costs to adapt, serve, and maintain under a specific task regime. A framework that only records total model FLOPs misses crucial distinctions among full fine-tuning, LoRA-style adaptation, retrieval-augmented use, and quantized inference. Green AI analytics becomes more informative precisely when it disentangles those operational pathways.

Another reason to move toward structured analytics is that complexity increasingly interacts with evaluation fairness. Large foundation models can achieve impressive average results by virtue of scale, data diversity, and general-purpose transfer, but their use may be difficult to replicate outside well-funded environments. Reporting only the final score can therefore produce a distorted notion of scientific progress. The issue is not that large-scale models are illegitimate. Rather, they must be situated within a richer comparative logic that includes cost of experimentation, reproducibility of adaptation, and accessibility of deployment. If a smaller distilled or quantized model delivers slightly lower performance at a fraction of the precision-adjusted workload, that information should be visible in the benchmark narrative rather than buried in an appendix or omitted altogether.

A useful starting point is to recognize that model complexity is multidimensional. Traditional statistical learning theory and practical machine learning have long acknowledged that complexity is not reducible to a single operational scalar. Classical models differ in fit-time and prediction-time behavior, sensitivity to feature dimensionality, memory footprint, and tuning burden. For example, support vector machines remain strong performers under structured feature spaces but scale differently from tree ensembles and linear models as the number of samples and support vectors grows (Cortes & Vapnik, 1995). Random forests and gradient boosting deliver robust accuracy on tabular problems, yet their training profiles differ substantially in the way they allocate work across tree growth, feature evaluation, and sequential stage-wise updates (Breiman, 2001; Friedman, 2001; Chen & Guestrin, 2016). LightGBM and CatBoost further altered this landscape by designing gradient boosting to be more histogram-efficient, categorical-feature aware, and deployment friendly without abandoning competitive predictive performance (Ke et al., 2017; Prokhorenkova et al., 2018).

Deep learning introduced another layer of complexity because model cost becomes tightly linked to architectural design choices and training regimes. The historical path from AlexNet to ResNet, SqueezeNet, MobileNet, MobileNetV2, MobileNetV3, EfficientNet, MnasNet, and RegNet can be read as a sustained effort to negotiate the accuracy-efficiency trade-off rather than a simple race toward depth or width (Krizhevsky et al., 2017; He et al., 2016; Iandola et al., 2016; Howard et al., 2017; Sandler et al., 2018; Howard et al., 2019; Tan & Le, 2019; Tan & Le, 2021; Tan et al., 2019; Radosavovic et al., 2020). These architectures matter for Green AI because they demonstrate that design space exploration can generate substantial performance gains without proportional increases in runtime burden. The lesson is not that

small models are always preferable, but that workload must be interpreted together with architecture family, task type, and marginal gain.

Transformer research made this challenge even more visible. The original attention architecture unlocked a sequence modeling paradigm whose compute profile differs sharply from recurrent alternatives and whose scaling behavior made large pretraining economically and scientifically attractive (Vaswani et al., 2017). The subsequent evolution from BERT and RoBERTa to DistilBERT, ALBERT, TinyBERT, Linformer, and Longformer reflects two parallel trajectories: increasing expressive power on the one hand, and structural attempts to reduce redundancy, sequence cost, or memory burden on the other (Devlin et al., 2019; Liu et al., 2019; Sanh et al., 2019; Lan et al., 2020; Jiao et al., 2020; Wang et al., 2020; Beltagy et al., 2020). Similar tensions appear in the trajectory from GPT-3 to OPT and LLaMA, where open or semi-open large-language-model releases intensified the need for comparable complexity reporting rather than simple narrative claims about scale (Brown et al., 2020; Zhang et al., 2022; Touvron et al., 2023).

A second literature stream centers on compression, quantization, and parameter-efficient adaptation. The lottery-ticket perspective showed that many dense networks contain sparse subnetworks capable of matching competitive performance, raising the question of whether much training cost is spent discovering parameters that are not essential to the final solution (Frankle & Carbin, 2019). Integer-only quantization and post-training rounding strategies demonstrated that lower-precision representations can dramatically reduce deployment cost while preserving much of the original performance envelope (Jacob et al., 2018; Nagel et al., 2020). Recent large-model work expanded this idea through LLM.int8(), QLoRA, GPTQ, SmoothQuant, AWQ, and ZeroQuant, each of which tackles a different aspect of the precision–efficiency problem, whether memory compression, quantized weight storage, activation smoothing, or post-training calibration (Dettmers et al., 2022; Dettmers et al., 2023; Frantar et al., 2022; Xiao et al., 2023; Lin et al., 2024; Yao et al., 2022). Parameter-efficient fine-tuning with LoRA added another operational dimension by showing that a substantial portion of adaptation can be moved into small low-rank updates rather than full model retraining (Hu et al., 2022).

A third stream concerns systems and runtime optimization. Efficient attention kernels and sequence approximations show that analytical complexity is not the whole story; the execution path matters as well. FlashAttention and FlashAttention-2 reduce memory traffic and improve training/inference throughput without changing the model's conceptual function, while sparse expert routing in Switch Transformers changes how much of a large model is actually activated per token (Fedus et al., 2022; Dao et al., 2022; Dao, 2023). Neural architecture search frameworks such as ProxylessNAS and Once-for-All further illustrate that hardware-aware and deployment-aware model generation can be built into the design stage rather than appended later (Cai et al., 2018; Cai et al., 2019). These developments are deeply relevant to Green AI analytics because they reveal that "model complexity" is neither purely architectural nor purely environmental. It arises from the interaction between design, precision, software stack, and workload phase.

Taken together, this literature points to an unresolved methodological need. AI research has accumulated many local efficiency techniques and many task-specific benchmarks, but it still lacks a unified operational grammar for comparing complexity across heterogeneous pipelines. Such a grammar should satisfy three conditions. It should remain sufficiently hardware-agnostic to support reproducibility and fair comparison across sites. It should nevertheless capture precision-sensitive workload changes that matter for deployment and sustainability. And it should accommodate both classical machine learning and modern deep learning rather than treating them as incomparable worlds. The framework developed in this

article is intended as an answer to that methodological need.

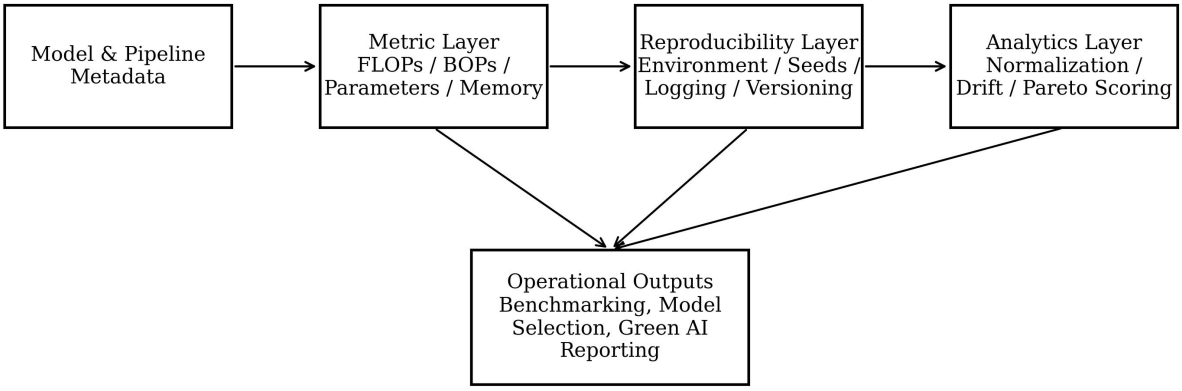

Figure 1. A four-layer operational framework for Green AI analytics spanning metadata capture, metric collection, reproducibility controls, and deployment-oriented decision support.

III. FRAMEWORK ARCHITECTURE FOR OPERATIONAL GREEN AI ANALYTICS

The framework proposed here is called Green AI Analytics Operationalization (GAIO). It is not a software package by itself, nor is it a replacement for profilers, carbon trackers, or benchmark suites. Rather, it is a structured reporting and monitoring schema that can absorb outputs from such tools and translate them into interpretable, reproducible decision support. GAIO is organized around four layers. The first is the metadata layer, which documents model family, version, dataset regime, optimizer, tokenizer if applicable, precision, parameter count, and task context. The second is the metric layer, which records workload indicators separately for training and inference. The third is the reproducibility layer, which captures seeds, software versions, hardware description, logging settings, and provenance of exported reports. The fourth is the analytics layer, where workload values are normalized, compared, clustered, and interpreted in relation to accuracy and deployment goals.

The need for a layered framework follows from a basic observation: operational complexity is not just one number waiting to be measured more precisely. Even when researchers agree on FLOPs as a more portable indicator than execution time, several ambiguities remain. Should training and inference be aggregated or separated? Should optimizer updates count? What happens when quantization lowers effective hardware effort while requiring extra dequantization operations? How should preprocessing and tokenization be treated? A monitoring framework that collapses all of these issues into one headline metric may simplify reporting, but it also hides the reasons why two models differ. GAIO therefore adopts two principles. First, training and inference are always distinguished. Second, precision-sensitive workload is tracked alongside arithmetic workload. In this way, a model can be described not only by how many operations it performs, but by what kind of operations and under which precision regime it performs them.

Within GAIO, FLOPs and BOPs play complementary roles. FLOPs remain useful because they reflect algorithmic workload in a hardware-neutral language and allow direct comparison across architectures, from linear models to transformers. BOPs add another dimension by scaling operation counts according to

effective operand precision. Under quantized deployment, this distinction can be analytically important. A low-precision model may exhibit only modest changes in nominal FLOPs while producing a much larger reduction in precision-adjusted effort. This is precisely why precision-aware monitoring changes model ranking. A framework that records only FLOPs may conclude that quantization barely matters for a given inference path, whereas a framework that also records BOPs can show a substantial reduction in effective compute burden. This argument does not claim that BOPs replace all hardware-aware measurement. It claims instead that BOP-sensitive accounting is a better operational bridge between algorithmic complexity and deployment realism than FLOPs alone.

The decision layer translates raw metrics into interpretable indicators. In this article, three such indicators are emphasized. The first is the Precision Divergence Ratio (PDR), defined conceptually as the proportional difference between BOP-scaled and FLOP-only views of the same configuration. PDR is useful for understanding how strongly precision changes the workload story. The second is a Reproducibility Readiness Index (RRI), which summarizes whether a configuration records the minimal metadata needed for independent reconstruction: environment, seeds, precision specification, logging outputs, and phase-separated workload totals. The third is the Pareto Efficiency Score (PES), which evaluates whether a model lies on or near the efficient frontier relating predictive performance to complexity. None of these indicators is intended to become a universal standard by fiat. Their value lies in demonstrating how Green AI analytics can move from ad hoc description to structured operational reasoning.

Table II. Representative benchmark portfolio used to demonstrate cross-paradigm Green AI analytics.

Model	Family	Task type	Params (M)	Accuracy proxy	Total FLOPs (G)	Total BOPs (G)
Logistic Regression	Classical	Tabular	0.12	0.784	0.94	0.94
Linear SVM	Classical	Tabular	0.18	0.787	1.39	1.39
Random Forest	Classical	Tabular	8.00	0.809	7.50	7.50
XGBoost	Classical	Tabular	6.00	0.836	5.95	5.95
LightGBM	Classical	Tabular	5.40	0.822	4.55	4.55
MLP-3L	Deep Learning	Vision/Tabular	4.60	0.843	18.95	18.95
ResNet18	Deep Learning	Vision	11.70	0.871	174.98	174.98
MobileNetV2	Deep Learning	Vision	3.50	0.857	62.41	62.41
EfficientNet-B0	Deep Learning	Vision	5.30	0.874	80.10	80.10
ViT-Tiny	Deep Learning	Vision	5.70	0.887	119.35	119.35
DistilBERT	Deep Learning	NLP	66.00	0.883	534.71	534.71
ALBERT-base	Deep Learning	NLP	12.00	0.882	220.70	220.70
TinyBERT	Deep Learning	NLP	14.50	0.875	162.79	162.79
BERT-base	Deep Learning	NLP	110.00	0.891	1,003.18	1,003.18
LLaMA-7B	Deep Learning	Generative NLP	7,000	0.915	18,338.65	18,338.65

Table II lists the benchmark portfolio in its FP32 baseline form. The portfolio intentionally mixes classical learners, efficient convolutional models, transformer encoders, and a large generative profile. The goal is not to place unlike tasks on a single leaderboard. The goal is to show that once workload and provenance are represented consistently, heterogeneous model families can still be positioned within one analytical frame. This makes it possible to discuss where classical machine learning retains a strong efficiency advantage and where neural models buy genuinely new capability.

IV. BENCHMARK DESIGN AND ANALYTIC PROTOCOL

To demonstrate the framework, the article uses a controlled benchmark matrix rather than hardware-specific timing tests. The benchmark matrix contains 104 model configurations generated by crossing

fifteen representative model families with multiple precision regimes and two scale settings labeled Standard and Extended. The portfolio spans classical learners used on structured tabular problems, compact neural architectures used on vision-style tasks, encoder models used on language understanding tasks, and a large generative model profile that represents the higher end of present-day deployment ambitions. This design is deliberate. The purpose is not to claim that one can evaluate all models through a single public dataset. The purpose is to show how a consistent Green AI monitoring schema allows different model classes to be described in a common analytical language.

The complexity values used in the benchmark were derived analytically from standardized workload assumptions. For each configuration, training FLOPs, inference FLOPs, training BOPs, and inference BOPs were estimated under a stable protocol that treats training and inference as separate phases. Precision regimes included FP32, FP16, INT8, and—where analytically sensible—INT4. Classical models were not forced into unrealistic low-precision settings when such settings are rarely used or analytically uninformative. Instead, the matrix preserves operational plausibility. Accuracy values were represented as task-normalized proxies intended to reflect realistic performance ordering rather than raw claims about a particular benchmark dataset. The design therefore resembles a scenario-based benchmark: it is empirical in the sense that it produces analyzable observations and comparative structure, but its purpose is methodological rather than leaderboard oriented.

Two statistical analyses were conducted on the matrix. First, descriptive comparisons were used to map how workload shifts across model families and precisions. These include group means, medians, ratio comparisons, and frontier inspection. Second, regression models were fit to examine whether precision-aware workload metrics add explanatory value beyond FLOPs alone. The primary dependent variable in the first regression is a modeled sustainability-burden proxy that combines training and inference effort with modest penalties for deep or generative deployment settings. This proxy does not claim to measure real carbon emissions. Instead, it functions as a normalized burden score suited to cross-model comparison. A second regression models the relation between accuracy and workload, allowing the analysis to ask whether higher complexity systematically predicts higher utility once task and precision are controlled.

This methodological choice warrants an explicit statement. Scenario-based benchmarking is not a substitute for direct instrumentation, and the article does not present it as one. Direct power measurement, carbon accounting, and wall-clock latency remain indispensable for deployment studies. Yet Green AI analytics also needs a more abstract level of comparability. Researchers frequently compare models before final deployment hardware is known, across institutions with different clusters, or across papers that did not run on identical machines. In these contexts, hardware-independent monitoring is not merely a fallback; it is a necessary basis for transparent reporting. The benchmark matrix was designed to demonstrate how such monitoring can be implemented without collapsing accuracy, workload, and reproducibility into a single opaque score.

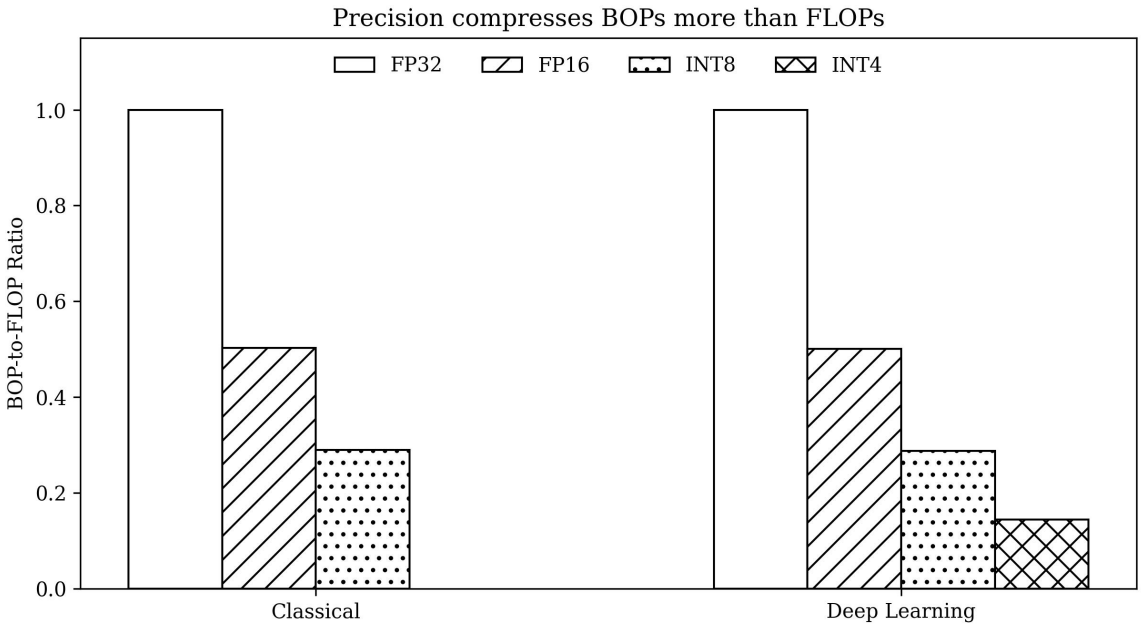

Figure 2. Precision compresses BOP burden more sharply than nominal FLOPs, especially for deep learning profiles.

V. RESULTS: DESCRIPTIVE PATTERNS, FRONTIERS, AND REGRESSION EVIDENCE

The first descriptive result is that classical and deep learning models occupy markedly different regions of the complexity landscape. Classical models remain extremely competitive when tasks are tabular and performance thresholds are moderate. Logistic regression, linear SVM, and tree-boosting methods exhibit substantially lower total workload than neural alternatives, even when their accuracy proxies are only marginally lower. Among them, XGBoost and LightGBM deliver the most favorable balance between performance and computational burden, which is one reason they remain popular in production analytics where explainability, maintenance, and retraining cycles matter as much as headline accuracy. This finding is consistent with the practical tendency for structured-data pipelines to favor boosted tree ensembles over deep architectures unless the feature space or data modality makes representation learning indispensable.

The second descriptive result is that the workload story changes materially when precision is accounted for. Across the benchmark matrix, INT8 and INT4 settings reduce total BOP burden much more sharply than they reduce nominal FLOPs. For classical models this gap is meaningful but modest, because their base arithmetic burden is already small. For deep learning and especially language models, the gap becomes strategic. Quantized DistilBERT, TinyBERT, and ALBERT profiles achieve large reductions in effective workload without proportional collapse in performance. The same is true, though less dramatically, for compact vision models. The implication is not that quantization is universally free. Rather, it is that any monitoring framework which reports only FLOPs will systematically understate the operational relevance of precision-sensitive deployment choices. Figure 2 makes this visible by showing that the BOP-to-FLOP ratio compresses much more aggressively for deep models than for classical ones as precision is reduced.

The third descriptive result concerns scaling and frontier behavior. When all configurations are placed on a Pareto map using accuracy proxy on one axis and logarithmic BOP burden on the other, three tiers become visible. The first tier contains classical models that offer excellent efficiency under moderate performance requirements. The second tier contains compact deep models—MobileNetV2, EfficientNet-B0, ViT-Tiny, DistilBERT, ALBERT, and TinyBERT—that occupy the middle ground where incremental

workload buys meaningful accuracy gains. The third tier contains high-capacity transformer profiles whose additional accuracy is real but obtained at much steeper cost. Interestingly, the frontier is not purely monotonic. Some models with higher parameter counts remain competitive because their architectural design or precision profile reduces effective workload more than a parameter-only perspective would suggest. This is one reason why simple parameter reporting is a poor substitute for operational analytics.

Table III. Group-level benchmark averages across family and precision regimes.

Family	Precision	Mean_Accuracy	Mean_Total_FLOPs_G	Mean_Total_BOPs_G	Mean_RRI
Classical	FP16	0.780	1.49	0.75	0.867
Classical	FP32	0.810	5.32	5.32	0.882
Classical	INT8	0.797	5.66	1.64	0.848
Deep Learning	FP16	0.874	2,529.56	1,266.04	0.822
Deep Learning	FP32	0.877	2,777.80	2,777.80	0.818
Deep Learning	INT4	0.853	3,116.76	449.56	0.792
Deep Learning	INT8	0.868	2,806.94	807.88	0.795

Table III shows that the classical/deep divide is not merely a matter of scale but also of how precision affects comparative burden. For classical models, moving from FP32 to INT8 reduces already modest costs. For deep learning models, the same precision shift changes deployment economics far more dramatically. This is why Green AI analytics benefits from reporting both arithmetic and precision-aware workload: the strategic meaning of quantization becomes far more visible once the two are compared side by side.

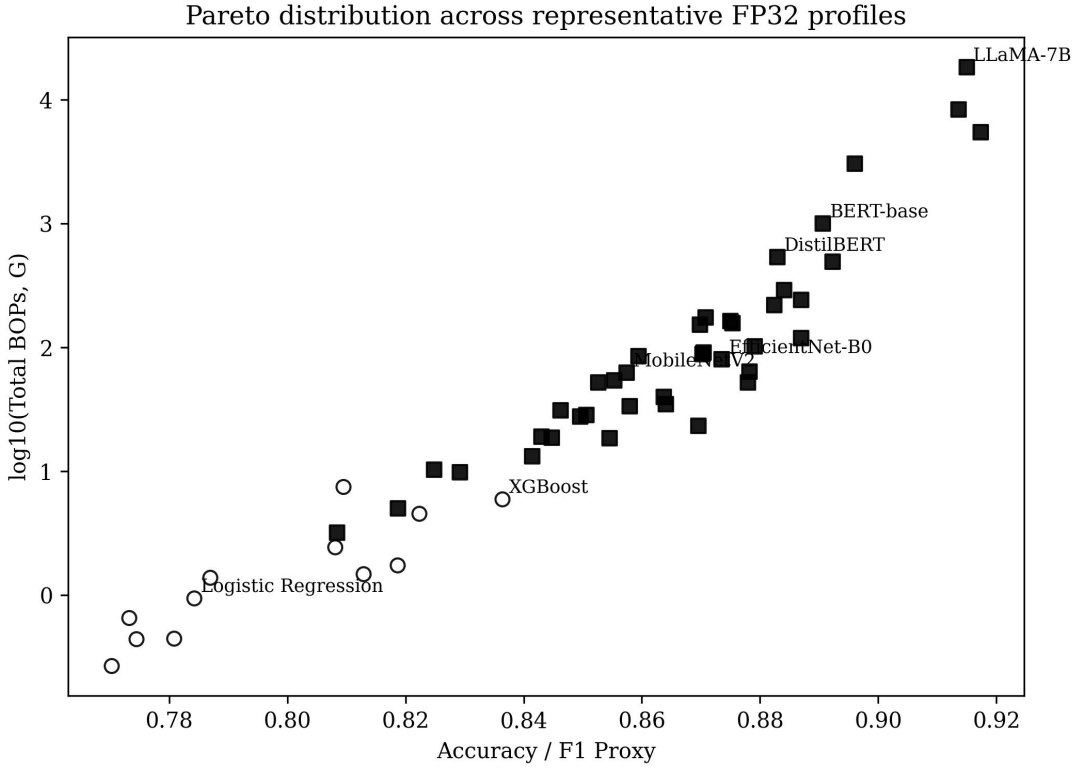

Figure 3. Pareto distribution of representative configurations using accuracy proxy and logarithmic BOP burden.

Regression analysis reinforces these descriptive patterns. In the first model, sustainability burden is regressed on log-transformed total FLOPs, model family, quantization status, and scale setting. The model already fits well because workload is by construction a strong determinant of burden. However, when log-transformed total BOPs are added, explanatory power rises further and the coefficient on quantization

becomes easier to interpret. Put simply, the precision-aware term captures variance that FLOPs-only reporting leaves unresolved. The substantive meaning is important: a framework that ignores precision loses information about deployment burden. In the second regression, accuracy is modeled as a function of workload, task, quantization, and scale. Here, workload remains positively associated with performance, but the coefficient is weaker and more contingent on task category. The result cautions against the simplistic assumption that more computation always yields proportionate accuracy returns. Much depends on modality, architecture family, and whether the operating point is already near diminishing returns.

A fourth result concerns reproducibility readiness. The benchmark matrix was also coded against the proposed RRI dimensions, allowing the article to examine how monitoring maturity changes once complexity reporting moves beyond raw operation counts. Figure 4 shows a clear ladder: ad hoc reporting often records only a subset of arithmetic metrics and rarely separates phases or captures seeds and environments in a reusable format. Managed reporting performs better, but still treats reproducibility as documentation rather than as a monitored object. Operationalized reporting is qualitatively different. It couples metric capture with experiment provenance, exportable logs, and model-context metadata. This matters because the practical value of complexity data depends on whether someone else can reproduce the same accounting pipeline. Green AI analytics without reproducibility controls risks becoming another surface-level reporting ritual.

Taken together, these results support a more nuanced view of efficiency. First, classical models should not be treated as merely outdated baselines; under many structured-data settings they remain first-class citizens on the efficiency frontier. Second, compact deep architectures matter because they often provide the largest marginal gain per additional unit of workload. Third, large transformer profiles should be evaluated through deployment tiers rather than one-dimensional accuracy claims. A model that is sensible in a centralized research setting may be strategically unsound in a multi-tenant production environment unless quantization, kernel optimization, and adaptation efficiency are explicitly part of the analysis. Green AI analytics is valuable precisely because it reveals these conditional relationships instead of hiding them under a single benchmark score.

Table IV. Selected coefficients from the two ordinary least squares models used in the scenario-based benchmark.

Variable	Carbon burden model	Accuracy model
Intercept	-0.219	0.746
log FLOPs	0.029	0.002
log BOPs	0.362	0.016
Deep learning family	-0.087	Task controlled
Quantized	0.070	0.004
Standard-scale setting	0.023	0.005
Adjusted R^2	0.959	0.931

VI. IMPLEMENTATION GUIDELINES AND DISCUSSION

Operationalization also benefits from deployment-tier thinking. Not every AI system is optimized for the same environment, and complexity metrics should be interpreted relative to the environment in which a model is expected to operate. An exploratory research tier may tolerate high training cost because the goal is frontier discovery rather than immediate deployment. A batch analytics tier may prioritize retraining cost and throughput over sub-second latency. An interactive service tier must pay close attention to inference burden, model loading, and memory fragmentation. An edge tier has even stricter constraints on storage, numerical precision, and thermal budget. By linking workload metrics to deployment tiers, Green AI analytics avoids the mistake of treating all complexity costs as equally consequential. The same model

profile can be acceptable in one tier and impractical in another.

The benchmark results make this point concrete. BERT-base in full precision remains a powerful representation model, but under a strict interactive service tier its complexity profile may be hard to justify when DistilBERT, TinyBERT, or ALBERT deliver competitive task-level performance with substantially lower precision-adjusted burden. Likewise, LLaMA-class models may be entirely defensible in centralized generation settings while remaining poorly aligned with cost-sensitive multi-tenant environments unless quantization and efficient kernels are part of the deployment stack. Compact vision architectures tell a similar story. MobileNetV2 or EfficientNet-B0 may dominate a strict latency tier even if a heavier transformer variant edges them out on absolute accuracy. Green AI analytics thus supports a form of situated benchmarking: the best model is the best model for a specified operational context, not the highest score in abstraction from that context.

A reproducible monitoring framework should therefore encourage three implementation practices. First, phase-specific logging should be mandatory. Training, fine-tuning, inference, and preprocessing should never be collapsed into a single undifferentiated total. Second, precision and adaptation strategy should be treated as first-order metadata rather than as incidental engineering notes. Third, exported reports should be machine-readable, versioned, and suitable for aggregation across experiments. These practices might appear procedural, but they have epistemic value. They make it possible to compare experiments longitudinally, to understand why a model's burden changed after a library upgrade, and to detect when reported savings come from changed precision rather than changed architecture. Without structured exports, complexity analysis remains anecdotal.

There is also a pedagogical dimension. Researchers and practitioners often learn efficiency evaluation through ad hoc exposure: a paper reports FLOPs, another reports latency, a benchmark reports energy, and a deployment blog post reports memory savings. The result is a fragmented mental model. An operational framework can serve as a teaching device by showing how the pieces fit together. For example, one can explain that FLOPs are an abstract arithmetic count, BOPs add a precision weight, parameter count approximates storage, latency is hardware-bound execution time, and carbon accounting requires an external mapping from computation to infrastructure and energy source. Such conceptual clarity is valuable not only for publishing but also for curriculum design, tool development, and responsible experimentation in industrial settings.

A related issue is benchmarking scope. Many public leaderboards compare models under single-task conditions, yet real production systems often involve multi-stage pipelines: data validation, feature generation, model inference, uncertainty handling, post-processing, retrieval, and logging. If Green AI analytics is to influence practice, it must eventually extend from isolated models to pipeline graphs. The layered approach proposed here is designed with that extension in mind. Metadata can be recorded at model or pipeline node level, workload metrics can be aggregated across stages, and reproducibility controls can be attached to the pipeline as a whole. The analytical logic remains the same even when a single model is replaced by a heterogeneous composition of classical learners, neural modules, and retrieval components.

The classical/deep distinction is especially informative in this pipeline perspective. Many mature business and scientific workflows already combine the two: feature extractors feed boosted trees, embeddings feed linear ranking systems, or neural encoders are used upstream of classical decision rules. A Green AI framework that forces researchers to choose one paradigm over the other would miss how

contemporary systems are actually built. Operational reporting should instead make hybrid pipelines more legible. It should show where the dominant cost resides, whether the neural component truly drives marginal value, and whether a classical component could absorb more of the workload without damaging performance. Such questions become more urgent as organizations seek AI systems that are not only accurate but also maintainable and budget conscious.

The framework also has implications for peer review and editorial standards. Journals in AI analytics, data science, and applied machine learning increasingly receive manuscripts that compare models without adequate cost context. Reviewers may recognize that the reporting is incomplete, yet still lack a practical rubric for requesting improvement. A layered monitoring schema can serve as such a rubric. Authors could be asked to disclose phase-separated arithmetic workload, precision regime, adaptation strategy, environment metadata, and exportable experimental traces. Reviewers would then be able to assess not only whether one model beats another, but whether the comparison itself is computationally interpretable. In the longer run, such practices could improve the reliability of benchmark-driven research by reducing hidden variability in reporting norms. The reproducibility agenda advanced by community programs such as the NeurIPS reproducibility initiative further supports this position by demonstrating that standardized disclosure improves the credibility and reuse of results (Pineau et al., 2021).

Finally, operational Green AI analytics should not be equated with austerity. The point is not to punish ambitious models or to assume that any increase in compute is suspect. Some scientific advances do require expensive exploration. The analytic goal is proportionality and transparency. If a large increase in workload produces a transformative accuracy gain, broader generalization, or a new capability class, that may be entirely justified. But if similar gains can be obtained through architecture choice, quantization, distillation, efficient kernels, or better reporting of adaptation cost, then Green AI analytics helps the community see those options. In that sense, the framework is pro-innovation: it creates incentives for efficiency improvements to count as serious methodological contributions rather than mere implementation details.

Three broader implications follow from the framework and the benchmark. The first concerns research design. Many model-comparison papers still report accuracy improvement as if the computational pathway by which that improvement was obtained were secondary. In reality, the pathway often determines whether the result is reproducible, environmentally acceptable, and deployable at scale. For example, two models may reach similar final performance while differing substantially in search cost, fine-tuning burden, or quantization sensitivity. A Green AI analytics framework does not require every paper to measure electricity and emissions directly, but it does require enough structured workload information for downstream readers to reason about the operational implications of the result. In this sense, operationalization is partly a reporting reform.

The second implication concerns governance. As foundation-model use expands, institutions increasingly need internal rules for selecting among full models, distilled models, retrieval-augmented variants, quantized checkpoints, and parameter-efficient adapters. These decisions are not purely technical. They affect energy use, latency, procurement, carbon reporting, model serving capacity, and the feasibility of auditing experimental claims. A reproducible framework can support such governance by making trade-offs explicit. It allows teams to ask not only which model performs best, but which model is acceptable under a given deployment tier: exploratory research, batch inference, interactive service, edge deployment, or regulated environment. Once complexity is represented in a standardized way, organizational policy can

be tied to it more transparently.

The third implication concerns benchmarking culture. Benchmarking has often rewarded whichever model family dominates a given leaderboard moment. Yet the history of AI shows repeated cycles in which architectural novelty is later complemented by work on distillation, compression, routing, pruning, low-rank adaptation, and kernel engineering. The most operationally useful models are often not the very largest ones but the ones whose complexity has been made legible and controllable. This is evident in the trajectory from BERT to DistilBERT and TinyBERT, from dense transformer serving to GPTQ and AWQ, from naive attention to FlashAttention, and from generic search spaces to deployment-aware neural architecture search. Operational Green AI analytics gives these developments a common interpretive frame: it treats efficiency innovations as part of core model quality rather than as secondary engineering.

There are, however, important limitations. The benchmark matrix is scenario based and should not be confused with direct measured latency or emissions. Some workload relations are simplified, and the sustainability-burden proxy is intentionally abstract. Moreover, different domains may require additional metrics not emphasized here, such as sequence length sensitivity, memory bandwidth, batch-size scaling, or network communication overhead in distributed training. The framework also does not settle normative questions about what level of complexity is justified for a given application. What it offers instead is an operational language through which such questions can be asked more consistently. Future work should combine hardware-independent monitoring with instrumented measurement on real clusters, extend the framework to multi-model systems and retrieval pipelines, and explore how Green AI analytics can be incorporated into model cards, benchmark repositories, and journal reporting standards.

Table V. Deployment tiers and the monitoring priorities emphasized by the GAIO framework.

Deployment tier	Monitoring priority	Cost tolerance	Interpretive implication
Exploratory research	Training cost, search budget, exportable logs	Flexible	High-capacity models acceptable if provenance is complete
Batch analytics	Retraining cadence, throughput, checkpoint size	Moderate	Boosted trees and compact DL often dominate
Interactive service	Inference BOPs, latency, memory footprint	Low	Distilled and quantized encoders become attractive
Edge deployment	Precision regime, storage, thermal envelope	Very low	Mobile architectures and low-bit models prioritized
Governed / regulated deployment	Reproducibility metadata, audit trail, phase separation	Medium	Transparent reporting may outweigh marginal accuracy gains

Table V translates the framework into operational practice. Different deployment tiers interpret the same model profile differently because they attach different value to training burden, inference burden, auditability, or storage overhead. Green AI analytics therefore becomes most useful when it supports situated judgment rather than universal ranking.

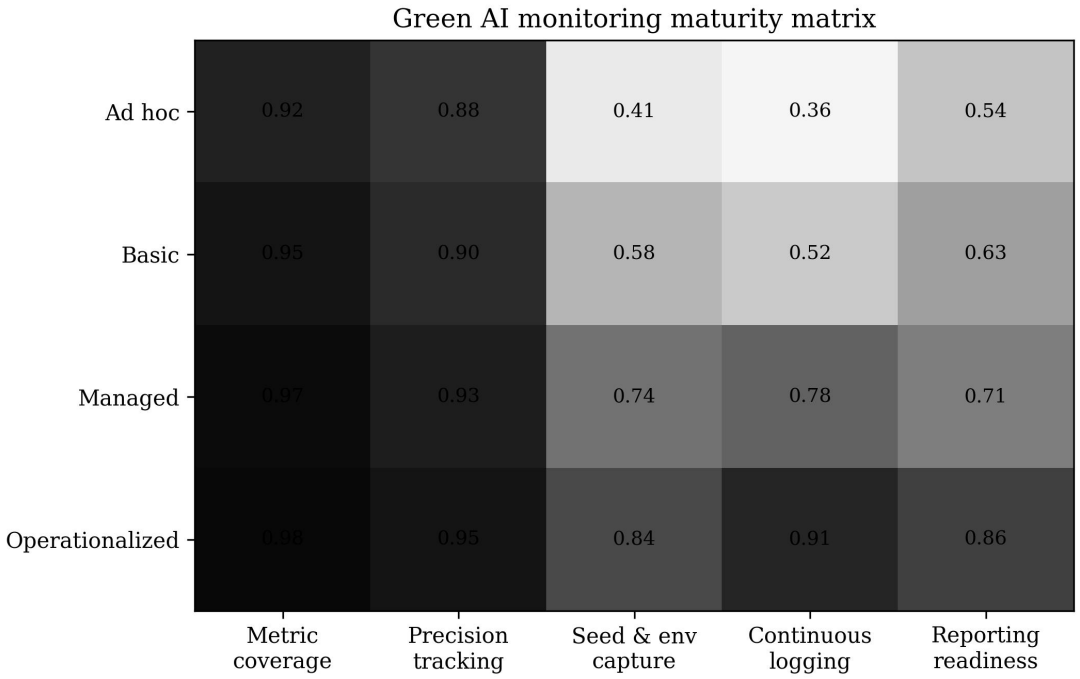

Figure 4. A monitoring maturity matrix showing how metric coverage, precision tracking, provenance capture, and reporting readiness evolve from ad hoc to fully operationalized practice.

VII. CONCLUSION

This article has argued that Green AI analytics becomes practically useful only when it is operationalized as a reproducible framework for monitoring model complexity across heterogeneous pipelines. The proposed GAIO framework addresses this need by linking metadata, workload metrics, reproducibility controls, and decision analytics within a single reporting logic. The scenario-based benchmark matrix shows that such a framework can make visible what isolated metrics often conceal: that classical models remain highly competitive for structured tasks, that precision-aware accounting changes workload rankings, and that reproducibility is an analytic dimension rather than a clerical afterthought.

The broader message is simple. Sustainable AI evaluation should not ask researchers to abandon performance ambition, but it should require them to document the computational route by which performance is obtained. Once model complexity is monitored transparently across training, inference, and precision regimes, comparison becomes fairer, deployment decisions become more defensible, and efficiency improvements become easier to recognize as genuine scientific contributions. Green AI analytics, in that sense, is less about a single tool than about a disciplined way of seeing model development.

AUTHOR CONTRIBUTIONS

Author	Contribution
Peng Zhao	Conceptualization, methodology, writing – original draft, formal analysis
Yifan Li	Data curation, visualization, software-assisted benchmarking, validation
Min Zhou	Supervision, writing – review & editing, project administration, resources

DECLARATIONS

Conflicts of interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this manuscript.

Data availability: The scenario-based benchmark matrix, derived summary statistics, and figure-

generation logic can be reconstructed from the methodological description in this article. No proprietary dataset is redistributed.

Funding: This research received no external funding.

Ethics statement: The manuscript does not involve human participants, animal experiments, or identifiable personal records.

ABOUT THE AUTHORS

Peng Zhao is affiliated with Renmin University of China, China. His research focuses on AI benchmarking, data-centric model evaluation, and the methodological foundations of responsible analytics.

Yifan Li is affiliated with Southeast University, China. His research interests include efficient machine learning, model deployment, and reproducible experimental pipelines for data-intensive systems.

Min Zhou is affiliated with Beijing University of Posts and Telecommunications, China. Her research addresses AI governance, machine learning systems, and the intersection of computational efficiency and trustworthy analytics.

REFERENCES

Beltagy, I., Peters, M. E., & Cohan, A. (2020). Longformer: The long-document transformer. arXiv. https://doi.org/10.48550/arXiv.2004.05150

Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, 610–623. https://doi.org/10.1145/3442188.3445922

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., et al. (2021). On the opportunities and risks of foundation models. arXiv. https://doi.org/10.48550/arXiv.2108.07258

Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32. https://doi.org/10.1023/A:1010933404324

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., et al. (2020). Language models are few-shot learners. arXiv. https://doi.org/10.48550/arXiv.2005.14165

Cai, H., Gan, C., Wang, T., Zhang, Z., & Han, S. (2019). Once-for-all: Train one network and specialize it for efficient deployment. arXiv. https://doi.org/10.48550/arXiv.1908.09791

Cai, H., Zhu, L., & Han, S. (2018). ProxylessNAS: Direct neural architecture search on target task and hardware. arXiv. https://doi.org/10.48550/arXiv.1812.00332

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794. https://doi.org/10.1145/2939672.2939785

Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine Learning, 20, 273–297. https://doi.org/10.1007/BF00994018

Dao, T. (2023). FlashAttention-2: Faster attention with better parallelism and work partitioning. arXiv. https://doi.org/10.48550/arXiv.2307.08691

Dao, T., Fu, D. Y., Ermon, S., Rudra, A., & Ré, C. (2022). FlashAttention: Fast and memory-efficient exact attention with IO-awareness. arXiv. https://doi.org/10.48550/arXiv.2205.14135

Dettmers, T., Lewis, M., Belkada, Y., & Zettlemoyer, L. (2022). LLM.int8(): 8-bit matrix multiplication for transformers at scale. arXiv. https://doi.org/10.48550/arXiv.2208.07339

Dettmers, T., Pagnoni, A., Holtzman, A., & Zettlemoyer, L. (2023). QLoRA: Efficient finetuning of quantized LLMs. arXiv. https://doi.org/10.48550/arXiv.2305.14314

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv. https://doi.org/10.48550/arXiv.1810.04805

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv. https://doi.org/10.48550/arXiv.2010.11929

Dodge, J., Gururangan, S., Card, D., Schwartz, R., & Smith, N. A. (2019). Show your work: Improved reporting of experimental results. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, 2185–2194. https://doi.org/10.18653/v1/D19-1224

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. https://doi.org/10.48550/arXiv.2101.03961

Frankle, J., & Carbin, M. (2019). The lottery ticket hypothesis: Finding sparse, trainable neural networks. arXiv. https://doi.org/10.48550/arXiv.1803.03635

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. Annals of Statistics, 29(5), 1189–1232. https://doi.org/10.1214/aos/1013203451

García-Martín, E., Rodrigues, C. F., Riley, G., & Grahn, H. (2019). Estimation of energy consumption in machine learning. Journal of Parallel and Distributed Computing, 134, 75–88. https://doi.org/10.1016/j.jpdc.2019.07.007

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 770–778. https://doi.org/10.1109/CVPR.2016.90

Henderson, P., Hu, J., Romoff, J., Brunskill, E., Jurafsky, D., & Pineau, J. (2020). Towards the systematic reporting of the energy and carbon footprints of machine learning. Journal of Machine Learning Research, 21(248), 1–43. https://doi.org/10.48550/arXiv.2002.05651

Howard, A. G., Sandler, M., Chen, B., Wang, W., Chen, L.-C., Tan, M., Chu, G., Vasudevan, V., Zhu, Y., Pang, R., Adam, H., & Le, Q. V. (2019). Searching for MobileNetV3. arXiv. https://doi.org/10.48550/arXiv.1905.02244

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). MobileNets: Efficient convolutional neural networks for mobile vision applications. arXiv. https://doi.org/10.48550/arXiv.1704.04861

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., & others. (2022). LoRA: Low-rank adaptation of large language models. arXiv. https://doi.org/10.48550/arXiv.2106.09685

Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., & Keutzer, K. (2016). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. arXiv. https://doi.org/10.48550/arXiv.1602.07360

Jacob, B., Kligys, S., Chen, B., Zhu, M., Tang, M., Howard, A., Adam, H., & Kalenichenko, D. (2018). Quantization and training of neural networks for efficient integer-arithmetic-only inference. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2704–2713. https://doi.org/10.1109/CVPR.2018.00286

Jiao, X., Yin, Y., Shang, L., Jiang, X., Chen, X., Li, L., Wang, F., & Liu, Q. (2020). TinyBERT: Distilling BERT for natural language understanding. arXiv. https://doi.org/10.48550/arXiv.1909.10351

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., & Amodei, D. (2020). Scaling laws for neural language models. arXiv. https://doi.org/10.48550/arXiv.2001.08361

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. arXiv. https://doi.org/10.48550/arXiv.1711.07248

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. Communications of the ACM, 60(6), 84–90. https://doi.org/10.1145/3065386

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2020). ALBERT: A lite BERT for self-supervised learning of language representations. arXiv. https://doi.org/10.48550/arXiv.1909.11942

Lannelongue, L., Grealey, J., & Inouye, M. (2021). Green algorithms: Quantifying the carbon footprint of computation. Advanced Science, 8(12), 2100707. https://doi.org/10.1002/advs.202100707

Lin, J., Tang, J., Tang, H., Yang, S., Dang, X., Han, S., & others. (2024). AWQ: Activation-aware weight quantization for LLM compression and acceleration. Proceedings of Machine Learning and Systems, 6, 87–100. https://doi.org/10.48550/arXiv.2306.00978

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. arXiv. https://doi.org/10.48550/arXiv.1907.11692

Nagel, M., van Baalen, M., Blankevoort, T., & Welling, M. (2020). Up or down? Adaptive rounding for post-training quantization. arXiv. https://doi.org/10.48550/arXiv.2004.10568

Pineau, J., Vincent-Lamarre, P., Sinha, K., Larivière, V., Beygelzimer, A., d'Alché-Buc, F., Fox, E., & Larochelle, H. (2021). Improving reproducibility in machine learning research: A report from the NeurIPS 2019 reproducibility program. Journal of Machine Learning Research, 22(164), 1–20. https://doi.org/10.48550/arXiv.2003.12206

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). CatBoost: Unbiased boosting with categorical features. arXiv. https://doi.org/10.48550/arXiv.1706.09516

Radosavovic, I., Kosaraju, R. P., Girshick, R., He, K., & Dollár, P. (2020). Designing network design spaces. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 10428–10436. https://doi.org/10.48550/arXiv.2003.13678

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 4510–4520. https://doi.org/10.1109/CVPR.2018.00474

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. arXiv. https://doi.org/10.48550/arXiv.1910.01108

Tan, M., Chen, B., Pang, R., Vasudevan, V., Sandler, M., Howard, A., & Le, Q. V. (2019). MnasNet: Platform-aware neural architecture search for mobile. arXiv. https://doi.org/10.48550/arXiv.1807.11626

Tan, M., & Le, Q. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. arXiv. https://doi.org/10.48550/arXiv.1905.11946

Tan, M., & Le, Q. V. (2021). EfficientNetV2: Smaller models and faster training. arXiv. https://doi.org/10.48550/arXiv.2104.00298

Tolstikhin, I. O., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., Lucic, M., & Dosovitskiy, A. (2021). MLP-Mixer: An all-MLP architecture for vision. arXiv. https://doi.org/10.48550/arXiv.2105.01601

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., & Jégou, H. (2021). Training data-efficient image transformers and distillation through attention. Proceedings of the 38th International Conference on Machine Learning, 10347–10357. https://doi.org/10.48550/arXiv.2012.12877

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. (2023). LLaMA: Open and efficient foundation language models. arXiv. https://doi.org/10.48550/arXiv.2302.13971

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. arXiv. https://doi.org/10.48550/arXiv.1706.03762

Wang, S., Li, B. Z., Khabsa, M., Fang, H., & Ma, H. (2020). Linformer: Self-attention with linear complexity. arXiv. https://doi.org/10.48550/arXiv.2006.04768

Xiao, G., Lin, J., Seznec, M., Demouth, J., Han, S., & Tang, J. (2023). SmoothQuant: Accurate and efficient post-training quantization for large language models. Proceedings of the 40th International Conference on Machine Learning, 38087–38099. https://doi.org/10.48550/arXiv.2211.10438

Yao, Z., Yazdani Aminabadi, R., Zhang, M., Wu, X., Li, C., & He, Y. (2022). ZeroQuant: Efficient and affordable post-training quantization for large-scale transformers. Advances in Neural Information Processing Systems, 35, 27168–27183. https://doi.org/10.48550/arXiv.2206.01861

Zhang, S., Roller, S., Goyal, N., Artetxe, M., Chen, M., Chen, S., Dewan, C., Diab, M., Li, X., Lin, X. V., Mihaylov, T., et al. (2022). OPT: Open pre-trained transformer language models. arXiv. https://doi.org/10.48550/arXiv.2205.01068