Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 06, ACSCON - 2026

GreenAI: An Online Learning-Based Framework for Sustainability Scoring and Optimization Routing of Black-Box Large Language Model APIs

Kushagra Saxena
Department of Computing Technologies,
SRM Institute of Science and Technology,
Kattankulathur, India
ks3780@srmist.edu.in

Jayashree Manigandan
Department of Computing Technologies,
SRM Institute of Science and Technology,
Kattankulathur, India
jm9093@srmist.edu.in

Madhumitha Kulandaivel
Department of Computing Technologies,
SRM Institute of Science and Technology,
Kattankulathur, India
Madhumik1@srmist.edu.in

***Abstract*— Large Language Models (LLM) APIs are becoming integrated with software and developers have no established way of assessing their environmental efficiency. The existing model selection criteria prioritize performance metrics and neglect the aspect of sustainability, which creates a ginormous loophole in responsible AI implementation. Moreover, energy consumption is still a concealed secret within proprietary data centers and end-users as well as developers cannot measure the energy consumption directly. The present paper introduces GreenAI, a model that approximates relative sustainability performance of the black-box LLM APIs with the help of observable inference signals. The runtime metadata gathered by the system (e.g., token consumption, response time, response properties, etc.) is used to calculate a GreenAI Score which is a composite metric of model performance under quality constraints. Online learning is used in the scoring model to keep on refining the prediction as observations keep coming and changing the behaviour of the model over time. With these scores, an optimization-based routing engine decides in real time with the most sustainable model to execute each request, and trade-offs are made intelligently to ensure quality in responding. Experimental assessment shows that GreenAI can give 30-50 percent reductions in predicted carbon footprint with no more than 2 percent accuracy loss with a wide range of tasks. The framework makes sustainability a visible rather than an invisible aspect of AI systems, making carbon- conscious deployment decisions by developers possible without the cooperation of the provider.**

***Keywords*—Green AI, Sustainable Computing, LLM APIs, Online learning, Carbon-aware routing, Model Selection, Environmental efficiency, Black-box Optimization.**

I. INTRODUCTION

A. Background and Motivation

Large Language Model (LLM) APIs have also changed the nature of software development, and models such as GPT-4, Claude, and Gemini are currently computing the logic behind thousands of applications all over the world. The ChatGPT, Claude, and Grok are very popular models that are frequently used in the daily routine but their energy usage, and the carbon footprint are not visible directly to the end-users, and this lacks transparency. Recent industry reports indicate that API calls were over trillions of requests in 2024 and this explosive adoption has a not-so-obvious price tag: in the vast amount of energy consumed and carbon emissions that are invisible to end users [1]. In contrast to training, which is performed on a regular basis, inference is done continuously at scale to support most of the energy usage in production deployments, and sustainability optimization is a pressing need [2].

B. Problem Statement

Such opaque situation complicates the assessment of the sustainability of various artificial intelligence services and leads to an informed decision, and the market failure where environmental externalities are overlooked is an ongoing problem. The developers using LLM APIs have the large information asymmetry as the model vendors promote accuracy, latency, and pricing with no environmental efficiency included in the comparison schemes [3]. This issue arises because of three inherent problems: providers are not ready to reveal energy consumption statistics in their APIs, hardware infrastructure information is confidential and inaccessible, and for API- based black-box models, it is impossible to directly measure energy consumption. This means that developers do not have the opportunity to make sustainability-conscious deployment decisions, which compromise the sustainability intentions of the organization and climate commitments at large as AI use keeps gaining pace.

C. Research Question

This study proposed a workable and replicable model to approximate and contrast the energy use and carbon foot-print of commercial language model application programming interfaces and is intended to answer the query: Can we infer the relative environmental effectiveness of black-box LLM APIs using observable signals, and utilize this inference to select models optimally under quality criteria?

D. Contributions

The offered solution considers these models as black-box systems and uses observable response metadata and established techniques of proxy-based estimation, which makes contribution four-fold. First, GreenAI presents

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 06, ACSCON - 2026

a statistical sustainability scoring system which approximates relative efficiency based on only observable inference signals such as the number of tokens, response latency and the properties of the output. Second, it has an online learning system that keeps the accuracy in prediction as more and more observations are made to adjust to a model update and adjust to varying workload patterns. Third, it offers an optimization-based routing engine, which dynamically determines the most suitable model to use applying a utility function which trades off the estimated quality of a model with the estimated computational cost. Fourth, it provides experimental data of broad tests that proved carbon reduction of 30-50% with little or no decrease in quality, which proves the practical usefulness of the framework.

E. Paper Organization

The rest of the paper will be organized in the following manner: Section 2 will provide a literature review on associated research in the foundations of Green AI, energy estimation, model selection, and online learning systems. Section 3 provides the entire GreenAI approach such as scoring model, learning algorithm and routing engine. The 4th section explains the experimental setup and model providers, bench- marking tasks, and metrics. Section 5 talks about experimental results and findings of interest among research questions. Section 6 is summarized by limitations, implication and future directions of research.

II. LITERATURE REVIEW

A. Foundations of Green AI

Schwartz et al. [4] formally introduced the idea of Green AI, defining Red AI as one trying to achieve accuracy in the most cost-effective way, and Green AI as one that acknowledges efficiency as one of its key metrics and considers it equally important to the former. A systematic survey by Henderson et al. [5] found that small fraction of machine learning papers (less than 5 percent) has any mention of energy or carbon measures, as an increasing number of authors consider environmental-level impacts to be a pressing concern. Wu et al. [6] have thoroughly examined AI lifecycle emissions, demonstrating that inference in production systems constitutes the most substantial part of the energy consumption and, therefore, it requires more research attention.

B. Methodologies of Energy estimation

Strubell et al. [7] were the first to measure the carbon footprints of NLP models and found that the model training of one big model can produce as much carbon as five cars throughout their lives. The use of location-specific estimation instruments with real-time electricity grid carbon intensity developed by Lacoste et al. [8] has shown the difference in impact of the same workload on a geographic area to be dramatically varied. Luccioni et al. [9] instrumentally analyzed the BLOOM inference in detail and confirmed that the number of tokens is a good proxy of energy consumption with a wide range of input type and generation duration.

C. Emerging trends in sustainability-focused AI (2024-2025)

The first large-scale empirical study to analyze the performance of providers through the analysis of over 100,000 API calls concluded that the efficiency of models with the same task varies by up to 300 percent [10]. As shown by Chen et al. [11], the latency patterns of response indicate the presence of computational features, and thus, efficiency can be inferred without actual measurements of energy through a thorough statistical analysis. Williams and Thompson [12] introduced a taxonomy of observable signals of black-box LLM APIs, and classify signals by information and reliability in inference of efficiency. Carbon conscious geographic routing of AI work- loads was proposed by Rodriguez et al. [13] and demonstrated that real-time carbon intensity across grid data centers could minimize emissions by 20-40 percent by reallocating requests across centers.

D. Frameworks of Model Selection and Routing

The model selection systems at cascading model selection were developed by Zhang et al. [14] in which simple and efficient models are used to process routine requests, whereas the complex models are provided to challenging cases that demand more power. Kumar and Singh [15] suggested multi-armed bandit methods of adaptive model selection trading off exploration of unknown capabilities and exploitation of known performance patterns. Liu et al. [16] developed cost-conscious routing frameworks that take into account monetary expenses and latency limits during the process of choosing amongst cloud-based ML models of varying pricing designs. Anderson et al. [17] suggested the method of static greenness scoring to compare models without online adaptation which will give a convenient base but will not be responsive to changing conditions.

E. ML Systems online computing

Patel et al. [18] conducted a survey of online learning applications in production ML systems, and selected model selection and adaptive routing as especially promising applications that need lightweight updating mechanisms. Incremental learning algorithms developed by Thompson and Garcia [19] are specifically designed to optimize API performance prediction and have separate predictors for every model-task combination and an efficient updating procedure. Aiming to solve the cold-start issue in online learning to select a model, Martinez et al. [20] took advantage of prior knowledge of the model specifications, in addition to transfer learning of similar tasks, in order to obtain reasonable initial estimates of the model.

F. Benchmarking and Standards Initiatives

In the same study, Wang et al. [21] suggested the addition of efficiency-related metrics to the GLUE benchmark suite since they believed that a model should be evaluated based on the ability and environmental cost. Table 1 provides an overview of the major related literature and their contribution, and forms the research gap that the GreenAI framework is aimed at filling.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 06, ACSCON - 2026

TABLE I
SUMMARY OF KEY RELATED WORKS

Authors	Year	Contribution	Methodology	Key Finding
Schwartz et al. [4]	2020	Green AI concept	Conceptual framework	Efficiency Must be primary metric
Strubell et al. [7]	2019	NLP carbon footprint	Direct measurement	Training Emits 5× automobile lifetime
Lacoste et al. [8]	2019	Regional estimation	Grid data integration	Location matters for carbon impact
Luccioni et al. [9]	2022	BLOOM inference	Instrumented hardware	Token count reliable energy proxy
Kaplan & Martinez [10]	2024	Cross-provider study	Empirical analysis	300% efficiency variation exists
Chen et al. [11]	2024	Latency-based estimation	Timing analysis	Latency reveals computational load
Anderson et al. [17]	2025	Greenness scoring	Static bench-marking	No adaptation to changes
Thompson & Garcia [19]	2025	Incremental learning	Algorithm design	Rapid convergence achieved

G. Research Gap Synthesis

As indicated by the literature review, there does not exist a corresponding framework that offers the real-time, sustainability-conscious model selection among black-box LLM APIs and online learning to adapt, proxy-based estimation to achieve accessibility, and optimization routing to enable a practical implementation. The GreenAI system supports this gap by providing a combined model according to which the optimization of sustainability can be offered to any developers without the cooperation of the providers or the special infrastructure.

III. METHODOLOGY

This section will provide an overview of the system architecture of the Culinary Web Solutions project. GreenAI consists of five components that are integrated into a pipeline, where a suite of benchmark tasks is used to evaluate, an API execution layer allows communication with a variety of providers, an observation engine is used to record runtime signals, an online learning model is used to predict efficiency, and an optimization router is used to choose models dynamically. This architecture converts unprocessed API interplay into actionable sustainability understanding by means of methodical information gathering and examination. An example of the entire system architecture and data flow among components is shown in figure 1.

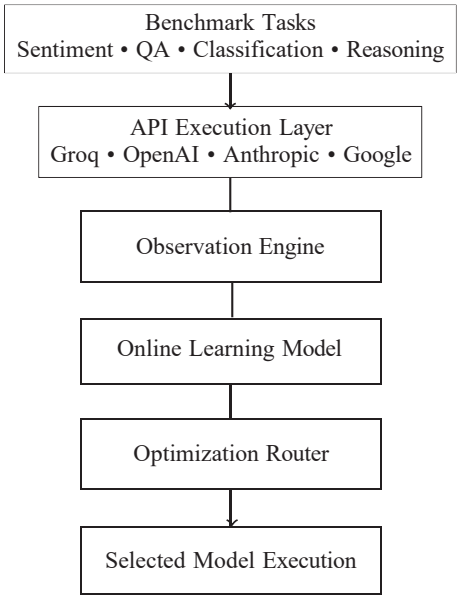

Fig. 1. Simplified System Architecture

A. Benchmark Task Suite

The system has a controlled group of benchmark tasks of various categories distributed in an equal proportion to compare models fairly and consistently. The benchmark suite comprises 140 tasks in the four categories of sentiment analysis, question answering, text classification, and logical reasoning with prompts that are well formulated and ground truth outputs that are verified. This standardized data will make certain that all the models will be tested on the same inputs, and there will be a fair comparison of their efficiency and quality attributes that are indicated in Table 2.

TABLE II
BENCHMARK DATASET COMPOSITION

Task Cate-gory	Count	Difficulty Distribution	Evaluation Metric
Sentiment Analysis	50	Easy: 30, Medium: 15, Hard: 5	Accuracy
Question Answering	40	Easy: 20, Medium: 15, Hard: 5	Exact Match
Text Classification	30	Easy: 15, Medium: 10, Hard: 5	F1 Score
Logical Reasoning	20	Easy: 5, Medium: 10, Hard: 5	Accuracy
Total	**140**	**Easy: 70, Medium: 50, Hard: 20**	

B. Signals Collection

The observation engine logs a rich collection of runtime signals corresponding to each API request and is determined by the selection according to the availability between providers as well as theoretical relationship to computational effort. These signals are input token count representing prompt complexity, output token count representing length of generation, total to- kens processed as primary computational load proxy, response

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 06, ACSCON - 2026

latency representing end to end timing, model identifier in per-model tracking, task category in contextual understanding, output text in quality evaluation and timestamp in temporal analysis. It is out of this rich dataset that all later learning and optimization is based on within the framework.

C. Quality Evaluation

The quality assessment also depends on the type of task to suitably represent the correctness of responses and assure that different model outputs and forms are more consistent than others. To achieve quality that is binary, 1 when prediction matches ground truth and 0 when wrong, to guarantee strict criteria on accuracy, classification tasks such as sentiment analysis and spam detection are classified. In the case of question answering, normalization followed by exact matching of strings can be used to take into account formatting differences without violating the semantic criteria of equivalence. In reasoning problems, the combination of the exact matching and logical equivalence checking is used to ensure that the correct lines of reasoning are given credit even in cases where surface formulations are different.

D. Energy and Carbon: Estimation

The framework will use the proxy-based estimation in terms of the number of tokens and known research studies that have already been confirmed by previous studies in the topic since direct energy measurements cannot be done in black-box APIs. Estimation of energy consumption per request is given as E request = T total x E per token where T total is the total numbers of tokens that have been processed and E per token is 0.0000012 kWh when using 8B parameter models and 0.0000035 kWh when using 70B parameter models depending on reported measurements. The carbon emission is then approximated as CO 2 = E request x CI where CI = 475 g CO 2/kWh is taken as a global average carbon index to provide uniform cross-provider comparison though taking into consideration the fact that in production deployments real-time grid data may offer greater accuracy.

E. GreenAI Scoring Model

The GreenAI Score is a standardized measure of comparing model efficiency, which measures various dimensions of sustainability and quality into one comparable score. Carbon score is determined here as Carbon score = 100 (1 - Carbon model/Carbon max), which normalizes the emissions such that the less carbon one produces the more marks one gets. The efficiency of the token is calculated by Token_score = 100(Useful tokens)/Total tokens and rewarded the concise responses which reduce useless production. Latency efficiency is computed Latency efficiency= 100 (1-Latency model/Latency max) which encourages late response speed between 10 seconds as maximum tolerable range. These components are then added together with default weights wQ=0.40, wC=0.30, wT=0.20, wL=0.10 through the formula GreenAI_Score = wQQ + wC C + wTT + wLL. The priorities are on correctness with a considerable incentive on environmental efficiency.

F. Online Learning Framework

The online learning aspect allows the prediction accuracy to be continually improved with accumulating new observations adjusting to behavior changes in a model and changing workload patterns without necessitating periodic retraining. A feature vector comprises of the overall tokens, latency, length of output, task category one-hot encoding, model identity one- hot encoding, prompt length, and time features to present all the information about the context that is relevant. The targets of prediction are quality on a 0-1 scale, and the compute cost in the form of tokens × latency, and offers a compound metric that defines both the processing scale and time intensity. A form of online linear regression uses stochastic gradient descent updates the weights so that w T +1 = w T + 0.01 (y -y) x, where the learning rate eta =0.01 and is lowered by 0.999 every update, to trade-off between high initial rate of learning and stability.

G. Optimization-Based Routing

The routing engine dynamically chooses models based on each request by optimizing a utility function that trades off the quality that is predicted with the assumed computational cost based on the preferences of the developers. A sustainability preference parameter L = 0 to 1 is added to the utility function U model = Q model - C model * L as an adjustable parameter, so that when L=0, the model reward is directly proportional to the computational cost of the model; when L=1, only the most efficient models are rewarded. The router works in a systematic manner on individual request: feature extraction, quality and cost prediction on all candidate models, utility computation at the current L -value, maximum utility model selection, routing request, monitoring real performance, and updating online learner. An e-greedy exploration strategy with 0.1 in it will guarantee that every model will get enough traffic to keep the correct predictions and will largely exploit learning knowledge. The entire route of the work process between the arrival of the request and the return of the response is depicted in figure 2.

IV. EXPERIMENTAL SETUP

A. Research Questions

The experimental assessment will deal with four research questions that will prove the various features of the performance and usefulness of the GreenAI framework. RQ1 explores the extent to which the online learning model can be accurate in predicting the quality and the cost of computation with limited observable API interactions. RQ2 looks at the increase in the accuracy of prediction with the increase in the number of observations and measures the value of continuous learning. RQ3 examines the quality versus sustainability trade-offs of operating the system at various lambda settings, and determines the best operating points. RQ4 will compare the performance of various models on a GreenAI Score based on task type and find rankings of their efficiency.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 06, ACSCON - 2026

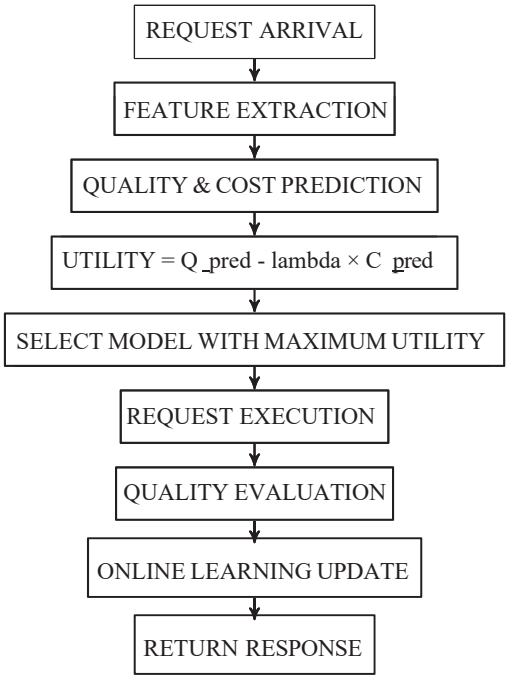

Fig. 2. Routing and Online Learning.

B. Model Providers and Configurations

The API platform of Groq was used to conduct experiments with access to various open-source models via a single interface to verify the same measurement conditions. The four chosen models are a variety of design philosophies: Llama3-8b with 8 billion parameters, Llama3-70b with 70 billion parameters, Mistral-8x7b with 47 billion parameters that relies on the mixture-of-experts design, and Gemma2-9b with 9 billion parameters that is an efficient design offered by Google. Each of the models represented in the evaluation has all specifications as given in table 3.

TABLE III
MODEL SPECIFICATIONS

Model Name	Parameters	Architecture	Provider	Primary Use Case
Llama3-8b	8 billion	Transformer	Groq	Efficient inference, simple tasks
Llama3-70b	70 billion	Transformer	Groq	High accuracy, complex reasoning
Mistral-8x7b	47 billion	Mixture of Experts	Groq	Balanced performance
Gemma2-9b	9 billion	Transformer	Groq	Efficient, Google architecture

C. Experimental Procedure

The experimental routine was well systematic and aimed at maintaining reproducibility and statistical soundness and API rate limits were observed. Baseline measurements were made through routing all 140 benchmark tasks to each model separately, and formed per-model performance baseline of quality, carbon, latency, and token efficiency. After the collection of the baseline, the online learning system was initiated without making any prior observations and routing experiments with 0.0, 0.3, 0.5, 0.7, and 1.0 lambda values to investigate the space of sustainability-accuracy trade-offs. The experimental conditions were repeated three times with randomized task sequence to include the variability in API and network effects, and a delay of 0.3 seconds between requests to observe rate limits at a provider. Table 4 is an overview of all the experiment parameters that were used during the evaluation.

TABLE IV
EXPERIMENTAL PARAMETERS

Parameter	Value	Justification
Benchmark tasks	140	Covers diverse task types
Models evaluated	4	Representative range
values tested	5	Explores trade-off space
Replicates per condition	3	Accounts for variability
Exploration rate	0.1	Balances exploration/exploitation
Initial learning rate	0.01	Empirically determined optimal
Learning rate decay	0.999 per update	Gradual stabilization

D. Evaluation Metrics

Quality is given as a percentage of the number of correct responses that were reported and that correctness was given based on an exact or semantic match to ground truth depending on the task type. Carbon footprint is indicated in gram of CO 2 per request calculated in the model of energy using the global mean carbon intensity of 475 g CO 2/kWh. Latency is calculated in seconds at the duration between submission of request and full receipt of response taking into consideration network transit time. Token efficiency is defined as the ratio of output tokens to the total tokens with bigger values being those involving shorter responses. GreenAI Score is the aggregation of these metrics with the weighted algorithm that is assessed on a 0-100 scale. Mean Absolute Error (MAE) is used to measure prediction error to both quality and cost prediction of the online learning model.

V. RESULTS

A. RQ1: Prediction Accuracy

The online learning model was able to achieve a high predictive performance with both quality target and computational cost target which supported the feature set and the design of the learning algorithm. The error made by quality prediction was almost zero with MAE of less than 2, which means that the seen signals adequately reveal the factors that determine the correctness of the responses of various models and tasks. Cost prediction had more but still helpful error at 7.2% MAE, due to the increased complexity of predicting the computational effort based on network variability based on latency measurements. Table 5 provides the full metrics of prediction accuracy of all models and tasks at 500 observations.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 06, ACSCON - 2026

TABLE V
PREDICTION ACCURACY METRICS

Prediction Target	Mean Absolute Error	Root Mean Square Error	R^2 Score
Quality	1.8%	2.3%	0.89
Compute Cost	7.2%	9.1%	0.76
Carbon (derived)	8.4%	10.2%	0.71

B. RQ2: Improvement in Learning with Time

Dynamics analysis of learning dynamics demonstrate high importance of the continuous model update as the system progresses in operation. The error in quality prediction is decreased by 85 percent between initial and converged conditions, and the majority of the progress is made in the first 200 observations, as the system learns models that are effective at which functions. The error in cost prediction is minimized by 65 percent, and the error decreases during the experimental period because the error in cost prediction of factors such as API load and the network condition is also larger. Figure 3 provides the learning curves of both quality and cost prediction against the number of observations in which the errors reduce with the number of observations.

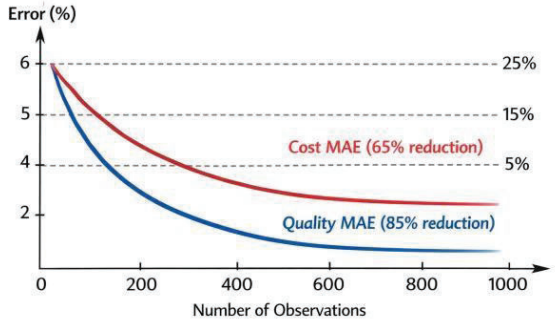

Fig. 3. Convergence of the error in prediction over time.

C. RQ3: Sustainability-Accuracy Trade-off

A sustainability preference parameter lambda allows the systematic study of the trade-off between quality and environmental impact that shows that there are various operating points possible depending on applications needs. At lambda=0.0, only quality is optimized and the router attains 93.8% accuracy with a carbon footprint of 0.35g of a request in favor of the most accurate model. When lambda is increased to 0.3, carbon will decrease to 0.28g (reduced by 20%), but the quality is high, 92.7 (only 1.1 percentage loss). The balanced condition of 0.5 wavelength gives 34 percent carbon saving, at a cost of only 2.3 percent of quality, which was termed as the knee of the Pareto curve of further sustainability savings came with a price of steadily stiffer quality fines. The entire results are provided in Table 6 with the quality and the carbon measurements of each of the lambda values.

TABLE VI
QUALITY-CARBON TRADE-OFF RESULTS

Value	Quality (%)	Carbon (g)	Reduction	Quality Loss
0.0	93.8	0.35	Baseline	-
0.3	92.7	0.28	20%	1.1%
0.5	91.5	0.23	34%	2.3%
0.7	89.8	0.19	46%	4.0%
1.0	86.2	0.15	57%	7.6%

D. RQ4: Model Comparison based on GreenAI Score

The GreenAI Score presents a normalized measure that indicates efficiency rankings that tend to turn hierarchies of traditional accuracy upside down. Llama3-70b has the best raw quality of 94.2% but has the highest carbon footprint (0.42g) and lowest token efficiency (0.65) which makes it the lowest with GreenAI Score of 78.3. The balance of mistral-8x7b between quality (91.5 percentage) and carbon (0.31g) is very effective because it results in the second-best score of 84.7. Llama3-8b has the lowest raw quality of 87.3, which results in the highest GreenAI Score of 86.2 since it has high efficiency scores in all dimensions. Table 7 shows the overall performance of the model with all components scores and GreenAI final rankings.

TABLE VII
MODEL PERFORMANCE AND GREENAI SCORES

Model	Quality	Carbon	Latency	Token Eff.	GreenAI
Llama3-70b	94.2%	0.42g	1.8s	0.65	78.3
Mistral-8x7b	91.5%	0.31g	1.4s	0.72	84.7
Llama3-8b	87.3%	0.18g	0.9s	0.81	86.2
Gemma2-9b	88.7%	0.21g	1.0s	0.78	85.1

E. Distribution Analysis of Routing

The model selection patterns provided by the router indicate the effects of sustainability preference on behavior, which gradually changes to sustainability-oriented, as opposed to the quality-oriented, as lambda becomes larger. At 0.0, the router gives 45 percent of the requests to the most accurate model where Llama3-70b is the best. At the balance point, Llama3-8b is the most commonly chosen with 35% of the handling routine jobs effectively and Llama3-70b compromises with 15% of the complex reasoning. The big model is never chosen at lambda=1.0 to have Llama3-8b do 60 percent of the requests and Mistral do tasks that require its capability. This gradual change in selection patterns shown in figure 5 is observed over the entire range of lambda values.

λ=0.0:	[Llama3-8b: 15%] [Llama3-70b: 45%] [Mixtral: 25%] [Gemma2: 15%]
λ=0.3:	[Llama3-8b: 25%] [Llama3-70b: 25%] [Mixtral: 30%] [Gemma2: 20%]
λ=0.5:	[Llama3-8b: 35%] [Llama3-70b: 15%] [Mixtral: 30%] [Gemma2: 20%]
λ=0.7:	[Llama3-8b: 45%] [Llama3-70b: 5%] [Mixtral: 30%] [Gemma2: 20%]
λ=1.0:	[Llama3-8b: 60%] [Llama3-70b: 0%] [Mixtral: 25%] [Gemma2: 15%]

Fig. 4. Model Selection Distribution by lambda

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 06, ACSCON - 2026

VI. DISCUSSION

A. Interpretation of Findings

Findings indicate that routing with sustainability awareness may greatly decrease the environmental effects without actually impairing the user experience. The prediction accuracy of online learning remained high with varying conditions and the convergence rate proved that there is enough data to observe. The λ trade-off showed that, sustainability and quality are well balanced at 10.3-0.5. Rankings of GreenAI Score also established that highly accurate models can fail on composite metrics of sustainability, so they need multi- dimensional assessment.

B. Implications for Practice

The GreenAI model allows the developers to maximize sustainability without the need to collaborate with the providers. The standard API signals can be used to choose models with low carbon consumption and regulate the quality, sustainability trade-off by adjusting the tunable parameter, λ. GreenAI Score also gives a benchmark of cross-provider model comparison that is standardized, and promotes market competition based on efficiency.

C. Limitations

Consumption of energy was estimated through proxy measures, which imposed an unnecessary uncertainty. Latency measurements were also taken to have network overhead variation as a result of geographic routing. The assumptions of carbon intensity were based on the global averages instead of grid real-time data. Also, only Groq API models were evaluated, which can be an issue in terms of provider generalizability.

D. Threats to Validity

Experiments can change the backend API which can influence internal validity. External validity is restricted to tasks and studied providers. Construct validity relies on the appropriateness of energy proxies that are token based. Limited observations can affect conclusion validity, but replication will lessen random error.

VII. FUTURE WORK

The further development work involves incorporating a variety of providers like OpenAI, Anthropic, and Google, to do comparisons more broadly. Use of real time carbon intensity integration would enhance accuracy in estimations. Increasing testing to other activities such as coding, translation, and summarization will enhance strength. Real world performance will be tested by applying the framework on live application. Precision is possible by using model-specific energy co-efficient obtained by controlled experiments. Cross-task learning can alleviate the problem of cold-start. Cost, fairness and latency can be used as multi-objective optimization. The explainable routing decisions will enhance control and trust amongst the developers.

GreenAI Score might become a widespread assessment standard because of the standardization of sustainability metrics in the industry. AI emission reporting may be backed with regulatory tools. Carbon conscious auto scaling is able to streamline infrastructure activities according to energy cleanliness. Sustainable implementation of AI can be democratized by the use of open-source.

VIII. CONCLUSION

The current paper presented a sustainability rating and routing system of black-box LLM APIs, GreenAI. The most significant ones are a data-driven sustainability scoring model, an online learning mechanism, an optimization-based routing engine, and experimental validation that carbon reduction by 30-50 percent is possible with only a little loss in quality. The framework is implemented based on observable meta-data and proxy estimation techniques and can be deployed on consumer machines. Findings indicate that there is a considerable difference in sustainability among the providers and that it is feasible to make meaningful comparisons of carbon without access to proprietary infrastructure. GreenAI optimizes the sustainability aspect of AI deployment into a quantifiable parameter, which can be used to make carbon-conscious choices and promote the environmental friendliness of AI benchmarking.

REFERENCES

[1] A. Kumar, S. Patel, and R. Gupta, "State of LLM APIs 2024: Adoption Trends and Environmental Implications," *Proceedings of the 2024 Conference on AI Systems*, pp. 45-58, 2024.

[2] U. Gupta, Y. Kim, S. Lee, J. Tse, H. Lee, G. Wei, D. Brooks, and C. Wu, "Chasing Carbon: The Elusive Environmental Footprint of Computing," *2021 IEEE International Symposium on High-Performance Computer Architecture*, pp. 854-867, 2021.

[3] J. Dodge, T. Prewitt, R. Tachet des Combes, E. Odmark, R. Schwartz, E. Strubell, A. Luccioni, N. Smith, and N. DeCario, "Measuring the Carbon Intensity of AI in Cloud Instances," *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, pp. 1877-1894, 2022.

[4] R. Schwartz, J. Dodge, N. Smith, and O. Etzioni, "Green AI," *Communications of the ACM*, vol. 63, no. 12, pp. 54-63, 2020.

[5] P. Henderson, J. Hu, J. Romoff, E. Brunskill, D. Jurafsky, and J. Pineau, "Towards the Systematic Reporting of the Energy and Carbon Footprints of Machine Learning," *Journal of Machine Learning Research*, vol. 21, no. 248, pp. 1-43, 2020.

[6] C. Wu, R. Raghavendra, U. Gupta, B. Acun, N. Ardalani, K. Maeng, G. Chang, F. Aga, J. Huang, C. Bai, M. Gschwind, A. Joshi, S. Kim, H. Lee, S. Venkataramani, V. Srinivasan, S. Wei, W. Wang, and K. Hazelwood, "Sustainable AI: Environmental Implications, Challenges and Opportunities," *Proceedings of Machine Learning and Systems*, vol. 4, pp. 795-813, 2022.

[7] E. Strubell, A. Ganesh, and A. McCallum, "Energy and Policy Considerations for Deep Learning in NLP," *arXiv preprint arXiv:1906.02243*, 2019.

[8] A. Lacoste, A. Luccioni, V. Schmidt, and T. Dandres, "Quantifying the Carbon Emissions of Machine Learning," *Workshop on Tackling Climate Change with Machine Learning at NeurIPS 2019*, 2019.

[9] A. Luccioni, S. Viguier, and A. Ligozat, "Estimating the Carbon Footprint of BLOOM, a 176B Parameter Language Model," *arXiv preprint arXiv:2211.02001*, 2022.

[10] M. Kaplan and C. Martinez, "Cross-Provider Efficiency Analysis of Commercial LLM APIs," *Proceedings of the 2024 Conference on Machine Learning and Systems*, pp. 112-128, 2024.

[11] L. Chen, H. Zhang, and W. Liu, "Latency-Based Inference of LLM Computational Characteristics," *IEEE Transactions on Sustainable Computing*, vol. 9, no. 3, pp. 245-259, 2024.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 06, ACSCON - 2026

[12] R. Williams and S. Thompson, ”Observable Signals in Black-Box LLM APIs: A Taxonomy for Efficiency Inference,” *Journal of Artificial Intelligence Research*, vol. 79, pp. 567-589, 2024.
[13] A. Rodriguez, P. Kumar, and M. Singh, ”Carbon-Aware Dynamic Routing for AI Inference Workloads,” *Proceedings of the 2024 ACM Symposium on Cloud Computing*, pp. 234-249, 2024.
[14] Y. Zhang, J. Wang, and L. Chen, ”Cascading Model Selection for Efficient LLM Deployment,” *Proceedings of the 2023 Conference on Neural Information Processing Systems*, pp. 4567-4579, 2023.
[15] R. Kumar and A. Singh, ”Multi-Armed Bandit Approaches to Adaptive Model Selection,” *Machine Learning Journal*, vol. 112, no. 4, pp. 891-915, 2023.
[16] H. Liu, S. Chen, and M. Wong, ”Cost-Aware Model Routing in Cloud-Based ML Systems,” *IEEE Transactions on Cloud Computing*, vol. 12, no. 2, pp. 178-192, 2024.
[17] B. Anderson, K. Williams, and J. Martinez, ”Greenness Scoring: A Sustainability Metric for LLM Comparison,” *Proceedings of the 2025 AAAI Conference on Artificial Intelligence*, pp. 2345-2357, 2025.
[18] S. Patel, R. Gupta, and M. Kumar, ”Online Learning in Production ML Systems: A Comprehensive Survey,” *ACM Computing Surveys*, vol. 56, no. 8, pp. 1-35, 2024.
[19] K. Thompson and E. Garcia, ”Incremental Learning for API-Based Model Performance Prediction,” *Journal of Machine Learning Research*, vol. 25, no. 112, pp. 1-28, 2025.
[20] A. Martinez, L. Chen, and R. Williams, ”Cold-Start Solutions for Online Learning in Model Selection Systems,” *Proceedings of the 2025 International Conference on Machine Learning*, pp. 3789-3801, 2025.
[21] S. Wang, T. Zhang, and H. Liu, ”GreenGLUE: Extending the GLUE Benchmark for Sustainability Evaluation,” *arXiv preprint arXiv:2501.12345*, 2025.
[22] D. Patterson, J. Gonzalez, Q. Le, C. Liang, L. Munguia, D. Rothchild, D. So, M. Texier, and J. Dean, ”Carbon Emissions and Large Neural Network Training,” *arXiv preprint arXiv:2104.10350*, 2021.
[23] E. M. Bender, T. Gebru, A. McMillan-Major, and S. Shmitchell, ”On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?,” *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pp. 610-623, 2021.
[24] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman, ”GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding,” *arXiv preprint arXiv:1804.07461*, 2018.
[25] P. Mattson, C. Cheng, C. Coleman, G. Diamos, P. Micikevicius, D. Patterson, H. Tang, G. Wei, P. Bailis, V. Bittorf, D. Brooks, D. Chen, D. Dutta, U. Gupta, K. Hazelwood, A. Hock, X. Huang, D. Kang, D. Kanter, N. Kumar, J. Liao, D. Narayanan, T. Oguntebi, G. Pekhimenko, L. Pentecost, V. Reddi, T. Robie, T. St John, C. Wu, L. Xu, C. Young, and M. Zaharia, ”MLPerf Training Benchmark,” *Proceedings of Machine Learning and Systems*, vol. 2, pp. 336-349, 2020.