E-ISSN:3108-1711
IRJSRR.ORG
INTERNATIONAL RESEARCH JOURNAL OF SCIENTIFIC REPORTS AND REVIEWS
Open Access | Peer Reviewed | Online Journal

Energy-Efficient Training of Large Language Models Through Sparse Attention and Low-Rank Adaptation (LoRA-S)

[*1]Anjani Kumar Tiwari; [2]Pawar Harish
[*1]CEO; Department of Civil Engineering, VIT University, Vellore Institute Of Technology, Vellore, Tamil Nadu; (632014).
[2]Senior AI/ML Engineer Department of Electronics & Communication Engineering VIT University, Vellore Institute Of Technology, Vellore, Tamil Nadu; (632014).
E-Mail: harish777pawar@gmail.com
[*]Correspondence Details: Email: anuj_gzp@yahoo.com

Abstract: The original purpose of large language models (LLMs) which include GPT and BERT has transformed natural language processing yet these models require extensive computational resources and energy consumption because of their enormous parameter counts and their quadratic attention system. The research presents LoRA-S as a comprehensive system which integrates Low-Rank Adaptation (LoRA) together with sparse attention technologies to enable energy-saving and effective training of transformer-based language models. The LoRA system lets users freeze pretrained weights while adding low-rank trainable matrices to both attention and feed-forward layers which results in over 90% trainable parameters reduction that decreases both gradient calculation needs and memory requirements. Sparse attention technology prevents tokens from interacting outside of defined groups which results in a reduction of attention-related FLOPs from 100 G to 7 G during WikiText-2 testing. The analysis of Full Fine-Tuning, LoRA, and LoRA-S shows that LoRA-S achieved the lowest energy consumption of 22,380 J (6.22 Wh) while displaying task performance which matched its competitors because it produced a perplexity score of 115.26 on WikiText-2 and achieved 73.90% accuracy on IMDB sentiment classification. The Pareto frontier analysis shows that LoRA-S provides the best balance between computational performance and forecasting capabilities while enabling environmentally sustainable model deployment in situations with limited resources. The study presents LoRA-S as a practical solution for Green AI because it combines two new methods that reduce FLOPs and parameter changes to maintain LLM performance. The research presents itself as experimental research which establishes LoRA-S framework to test and assess energy-saving methods for large language model training.

Keywords: *Large Language Models, Low-Rank Adaptation (LoRA), Sparse Attention, Energy-Efficient Training, Green*

Received: 06-05-2026 **Revised:** 11-05-2026 **Accepted:** 14-05-2026
Citation for the Paper:

1. Introduction

Large language models (LLMs) that include BERT, GPT, and their successors have basically changed the whole reality of artificial intelligence by getting the highest ratings ever in natural language comprehension, reasoning, and generation tasks [1]. These models are the basis of a great number of real-world applications, which comprise conversational agents, automated content generation, information retrieval, medical decision support, and intelligent control systems. The state-of-the-art performance of LLMs is mainly a result of significant model scaling, where contemporary architectures have hundreds of millions to hundreds of billions of parameters [2]. Nonetheless, this rapid scaling has come with a high price in terms of colossal challenges concerning computational

cost, memory usage, and energy consumption. Training and fine-tuning large language models can be very demanding, and thus, they are typically done with large-scale GPU or TPU clusters that operate for a long time. Take, for instance, the case of a transformer-based model with several billion parameters; it can easily take over a thousand GPU-hours to train, and thus the energy consumption involved would also be in the range of hundreds of MWh. Earlier research has indicated that the training of a single huge natural language processing model may lead to the emission of tens or even hundreds of tons of CO_2-eq., which is similar to the lifetime emissions of several passenger cars. Such estimates have made the issue of the environmental friendliness of large data deep learning techniques more acute, particularly when the use of LLMs is ever-growing in various sectors and research areas.

Moreover, besides the environmental concern, the CPU-intensive requirement of LLM training and fine-tuning is a major drawback [3]. Traditional fine-tuning methods involve making changes to all the parameters of the model, thus causing heavy memory bandwidth usage and a lot of data transfers between the memory and the computing units. In the case of a model with billions of parameters, full fine-tuning may need GPU memory of up to tens of gigabytes, which makes it impossible for a number of academic labs, small businesses, and edge-computing environments to use. Hence, the progress and customisation of LLMs are mainly taking place within organisations that have enormous computing power, thus hampering the development of new ideas and the conducting of reproducible research on a wider scale. The research community has increasingly highlighted energy-aware and parameter-efficient training techniques as a way to overcome the mentioned hurdles. In this regard, two major research paths have come to the fore: sparse attention mechanisms and parameter-efficient fine-tuning strategies. The goal of sparse attention techniques is to bring down the quadratic computational complexity involved in the self-attention operation that grows as $O(n^2)$ with an increase in sequence length. Sparse attention methods have the potential to bring down the computation noise to the levels of nearly linear or logarithmic scales by applying constraints on the attention computation to very particular subsets of the tokens, such as local windows, global tokens, or sparsity patterns that are structured. This cuts down on the computation, resulting in a great reduction of the number of floating-point operations (FLOPs), memory consumption, and energy usage, especially in long-context processing[4]. On the other hand, the parameter-efficient fine-tuning (PEFT) techniques have been suggested to prevent the entire model parameters from being updated during the process of adaptation. Amongst these approaches, the Low-Rank Adaptation (LoRA) method is recognized as the most efficient and popular technique that is widely used.

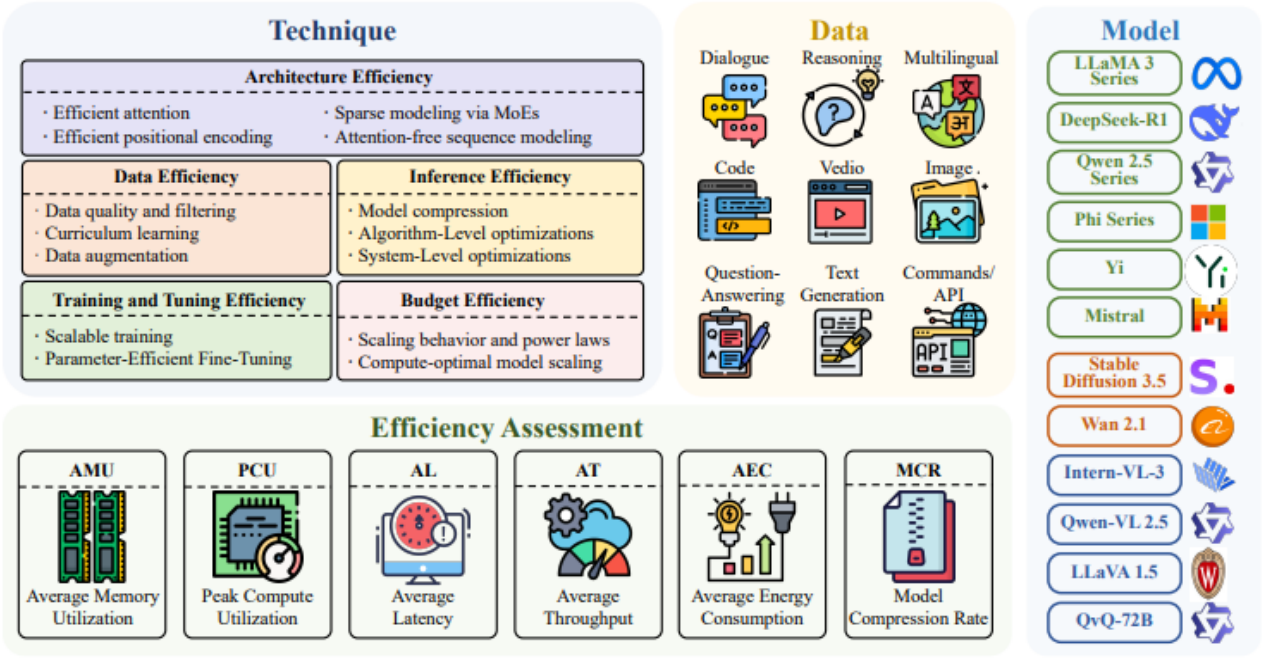

Figure 1: Overview of the Efficient LLM Framework [6].

In addition, one of the most important objectives is to test the proposed framework's scalability with various model sizes and sequence lengths so that the applicability of the framework to contemporary transformer architectures is assured. Moreover, the research is aimed at examining the possibility of the deployment of LoRA-S in resource-limited environments like single-GPU systems and edge-oriented platforms, which would result in the improvement of availability and sustainability. Finally, the research intends to make LoRA-S a practical move toward Green AI by showing that the energy as well as carbon footprints can be greatly reduced with the same model performance if the adaptability and generalisation are not compromised over the course of the transition. Figure 1 presents a comprehensive picture of the development of large language models (LLMs) that prioritise efficiency, showcasing the main techniques, data types, model families, and evaluation metrics. Besides, it classifies the efficiency gains according to different dimensions such as architecture, data, training, inference, and budget. At the same time, it portrays the variety of tasks and the different types of data, i.e., multimodal data, that are being processed by contemporary LLMs. Furthermore, the figure provides a quick reference to the principal metrics for assessing efficiency: memory usage, latency, throughput, energy consumption, and model compression. The figure noticeably entails the necessity of overall efficiency optimisation throughout the entire LLM lifecycle.

LoRA, low-rank adaptation, is a new method that consists of freezing the weights of the pretrained model and introducing low-rank trainable matrices to the selected layers, usually the attention and feedforward projections. In doing this, it is possible to reduce the number of trainable parameters

massively, up to more than 90%, resulting in significantly decreased memory and gradient computation costs while keeping the performance for the task at a competitive level. It has been shown with empirical studies that LoRA-based fine-tuning can end up with a performance level almost equal to that of the full fine-tuning, but only consuming a small fraction of the costs [5]. Sparse attention and LoRA are two approaches that have been successful individually but have mostly been treated as separate solutions. On the one hand, Sparse attention is mainly concerned with cutting down the attention-related FLOPs, while on the other, it works on the concept of lessened parameter updates and memory access. In the case of practical training pipelines, however, energy consumption is a result of the combination of attention computation and parameter updates. Not considering one component while optimizing the other restricts the possible efficiency gains. This situation points out an important lapse in the scientific literature: the lack of a common framework that takes simultaneously into account both the complexity of attention and the efficiency of parameters in the training of LLMs, sparking inspiration from this deficiency, the current study introduces LoRA-S, a single energy-efficient training framework that combines low-rank adaptation with sparse attention mechanisms. The main concept of LoRAs is to simultaneously lessen the two major factors that cause high LLM training costs: (i) the heavy computational load of self-attention and (ii) the number of parameters that can be trained along with the corresponding memory operations. By incorporating LoRA components within transformer architectures that use sparse attention, the proposed method leads to a combined reduction in FLOPs, memory bandwidth usage, and time required for the whole training process.

The primary objective of the present study is to create a training framework for large language models that is both energy-efficient and scalable by cutting down computational complexity, together with parameter update overhead. To be more precise, the present research aims to build a unified LoRA-S framework that will merge sparse attention mechanisms with low-rank adaptation in order to lower the number of floating-point operations, memory bandwidth usage, and overall energy consumption during the fine-tuning process. One of the main goals of the research is to analytically measure the computational and energy savings obtained by this integration, showing that the coming together of sparsity in both attention and parameter updates results in greater efficiency gains as compared with the traditional full fine-tuning and single-parameter-efficient methods.

2. Related Work

2.1 Energy-Aware Learning and Green AI

The enormous increase in the size of deep learning models has resulted in a corresponding rise in the demand for computation and the consumption of energy. Some researchers have already conducted studies in which the training of large neural networks has been systematically evaluated in terms of its

financial and environmental impact. The use of state-of-the-art NLP models was found to result in an energy consumption of hundreds of megawatt-hours, which in turn leads to considerable carbon emissions Usman, Y et al., 2025 [7]. Their results have pointed out that the performance improvements are frequently accompanied by disproportionately high environmental costs; thus, it is the research community that has been provided with an incentive to shift toward efficiency-oriented work. The author was inspired by this observation when Schwartz et al. formally presented and introduced the concept of Green AI, arguing and recommending that computer power use, energy consumption, and environmental impact should be regarded as and included in the top evaluation criteria, along with accuracy. This paper claimed that announcing only the highest performance metrics encourages the over-scaling of models, while the efficiency-aware benchmarks stimulate sustainable innovation.

However, subsequent later that studies' work within this viewpoint suggested and proposed the energy-normalized metrics such as energy per training step, FLOPs per task, and carbon-aware optimization strategies. In the case of large language models, it has become more and more the case that the principles of Green AI are relevant, given the huge modern transformer architectures. Recent surveys and position papers stress that if no fundamental changes are made in model design and training strategies, the environmental impact of LLMs might become untenable Shahzad, T. et al. 2025 [8]. The research has been unified by the common and universal findings, which together form a powerful impetus for the investigation of architectural and algorithmic approaches that can cut down training energy while maintaining model capability.

2.2 Sparse Attention Mechanisms

The self-attention mechanism, which is one of the major factors contributing to the high computational costs of transformer-based models, has its complexity that grows quadratically with the length of the sequence Choi, S. R. et al .2023 [9]. Various methods have been proposed to eliminate this problem, one of which is the sparse attention mechanism that allows the selection of token subsets for attention computation. The Sparse Transformer presented block-structured and strided attention patterns to eliminate the complexity of attention while still providing expressiveness, Sarpietro, R. E et al. 2022 [10]. Longformer, as well, came up with a hybrid attention mechanism consisting of local windowed attention and global tokens specific to the task, thus achieving near-linear complexity and allowing the long processing of documents in an efficient way. BigBird took a step further in the direction of random, local, and global attention combinations, thereby introducing a new way of working with sparse attention patterns, as well as guaranteeing the same level of expressive power as full attention in terms of theoretical conditions. Models using sparse attention achieved a great deal of reduction in FLOPs and memory required, especially in the case of long-context tasks like document classification

and question answering.

2.3 Parameter-Efficient Fine-Tuning

At the same time that sparse attention research was being conducted, the parameter-efficient fine-tuning (PEFT) methods, with their powerful prowess, and full model fine-tuning, as an alternative method to it, have grown together, Wang, L. et al., 2025 [11]. The traditional fine-tuning procedure updates all the model parameters, leading to high memory consumption and significant overhead caused by the computation of gradients. The PEFT strategies are intended to use only a few extra parameters to adapt the pretrained models while keeping the original weights frozen. The Low-Rank Adaptation (LoRA) method has come to the forefront among the PEFT methods due to its simplicity and power Lin, W. 2026 [12]. LoRA balances the low-rank matrices that can be trained into the weight projections to be changed, thus cutting down the number of parameters that can be trained by more than ten times. Performance studies indicate that the LoRA technique is just as good as the full fine-tuning one in terms of performance in a huge variety of NLP tasks while lowering the costs of memory and training significantly Taylor, N. et al., 2024 [13].

2.4 Limitations of Existing Approaches

Sparse attention and parameter-efficient fine-tuning, although separate and totally different techniques, come together to yield the highest efficiency since each of them gives a good share of the efficiency improvements independently Nwaiwu, S., 2025 [14]. Existing literature hardly mentions the interaction between these approaches, mostly decision-making based on each method's efficiency for its own sake. While sparse attention methods cut attention computations down to size, they often still use full parameter fine-tuning or optimization for inference only. On the other hand, LoRA and its related PEFT techniques substantially minimize parameter updates but still engage in dense attention operations that continue to eat up FLOPs and energy Han, Z., Gao et al. 2024 [15]. Besides, most of the literature has been driven by the indirect proxies for efficiency, such as training time or memory consumption, that do not place much focus on total energy-aware analysis to compute and parameter updates. Therefore, the current techniques cannot entirely leverage the collective capacities of artificial sparsity and parameter efficiency. Among the published works, the one that is the most evident is the absence of integrated frameworks that tackle the two highest contributors to LLM training cost, attention computation and parameter optimization, simultaneously Kumar, P., 2024; Tu, X. et al., 2024 [16-17].

Even though there have been considerable advancements in energy-efficient AI, sparse attention mechanisms, and parameter-efficient tuning techniques, current research still considers these facets

mostly apart. Green AI has been previously studied only in the manner of measuring energy consumption and suggesting efficiency metrics without advocating unified architectural solutions for large language models Fakhabi, M. M., Hamidian, S. M., & Aliehyaei, M., 2024; Barbierato, E., & Gatti, A., 2024 [18-19]. On the other hand, among the technological advancements in machine learning, sparse attention has been able to reduce self-attention's quadratic computational complexity, but not the energy overhead resulting from parameter updates and gradient computations that are still part of the training and fine-tuning processes. In contrast, techniques based on parameter efficiency, like LoRA and QLoRA, have significantly cut down on the number of trainable parameters, yet they still depend on the dense attention operations, which are the main source of both computational and energy cost in transformer models. Besides, most of the existing works discuss efficiency enhancement either at the architecture level or at the parameter level, failing to present a holistic framework that reduces attention computation, memory access, and gradient update costs simultaneously Cong, S., & Zhou, Y., 2023; Ahmed, S. F. et al., 2023 [20-21]. The lack of an integrated method that properly tackles both attention complexity and parameter efficiency is a major research gap, especially when considering the issue of sustainable and scalable training of large language models.

To fill the research gap that has been pointed out, the current study presents LoRAs, which is a unified and energy-efficient training framework that integrates the sparse attention mechanism with low-rank adaptation in a seamless manner. In contrast to previous solutions that were focused on either the improvement of attention computation or the parameter efficiency, LoRA-S does the opposite by reducing both the forward-pass attention FLOPs and the backward-pass gradient computation costs at the same time. The suggested framework directly incorporates the low-rank adaptation modules into the transformer layers that use sparse attention, resulting in a huge decrease in the number of parameters that are trainable, less memory access, and overall energy consumption. Besides, LoRA-S is not tied to any specific architecture and can be utilized with encoder-only, decoder-only, and encoder-decoder transformer models, thereby increasing its generality and scalability. The work does an analytical and empirical linking of computational complexity, parameter efficiency, and energy consumption concurrently, which is a major step taken towards the large-scale model training and towards the practical realization of Green AI principles, which the authors claim as a new contribution to the field of sustainable technology.

3. Research Methodology

The process flowchart depicted in Fig. 2 portrays the entire procedure of the suggested LoRA-S (Low-Rank Adaptation with Sparse Attention) system. To start with, a pre-trained transformer model is utilized, and the original weights are not updated to prevent the high computational and memory cost

that comes with full fine-tuning.

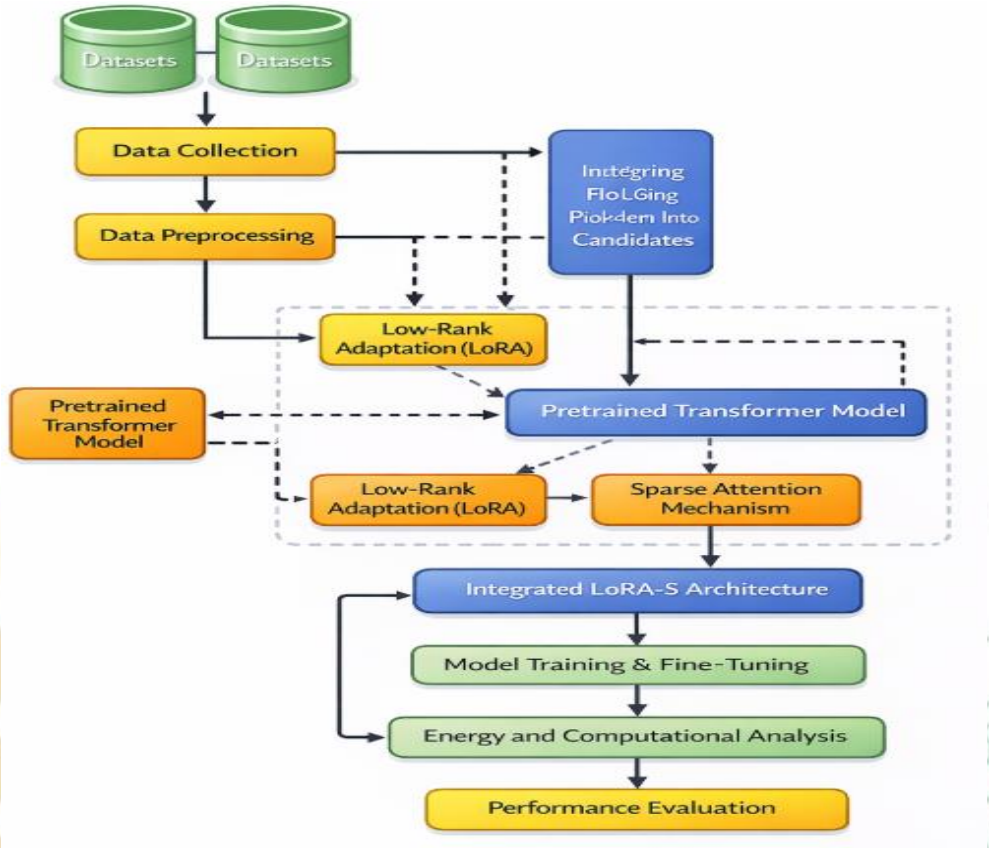

Figure 2: Overall Methodological Workflow of the Proposed LoRA-S Framework

The transformer layers are then split into attention and feedforward parts, where low-rank adaptation is put into practice to cut down the number of trainable parameters, and at the same time, sparse attention mechanisms are used to limit the attention computation to certain token groups, thus enhancing the overall computational and energy efficiency.

3.1 Data Collection

The selection of well-established and publicly available natural language processing benchmarks guaranteed reproducibility and a fair comparison with the existing studies. The datasets applied span a wide range of language understanding and generation tasks like text classification, question answering, and summarization. The study uses WikiText-2 as a language modeling dataset and IMDB as a sentiment classification dataset. The researchers chose these datasets to test various transformer training methods through their ability to process data efficiently and produce accurate results. The WikiText-2 dataset enables testing of extended sequence language models while IMDB provides a binary sentiment classification framework to evaluate model performance. The characteristics of the chosen datasets include varied sequence lengths, complexity of the language used, and different

semantic structures, which thus make it possible to evaluate the sparse attention mechanisms in both short-context and long-context conditions. All the datasets used were from open research repositories, and they maintain the so-called standard training-validation-test splits; this was done to make sure that the current study is consistent with previous research and to allow for objective performance benchmarking.

3.2 Data Preprocessing

All datasets were preprocessed in a standard way before model training; thus, textual data with a transformer-based architecture support was the outcome. Text normalization came first to eliminate encoding inconsistencies and achieve a common representation. The normalized text was tokenized using sub-word tokenization schemes such as Byte Pair Encoding (BPE) or WordPiece, which were aligned with the vocabulary of the pretrained model. The token sequences were padded or cut off to a firm maximum length, which allowed for efficient batch processing. During training, the attention masks were created to tell the valid tokens apart from the padding ones. Also, sparse attention patterns were created by specifying token subsets based on locality, global tokens, or specified sparse connectivity rules, thus making it possible to compute attention in an efficient manner. This preprocessing pipeline guarantees stable training, less memory utilization, and easier adaptation of sparse attention and low-rank through the proposed LoRA-S framework.

3.3 Model Framework

The suggested LoRA-S (Low-Rank Adaptation with Sparse Attention) system is aimed at concurrently cutting down (i) the parameter update cost in backpropagation and (ii) the attention-related computation load during both passes of forward and backward. By collectively optimizing these two main factors responsible for the costly training, LoRA-S realizes remarkable cuts in the energy consumed during the training period while the model is still performing at its best in the task.

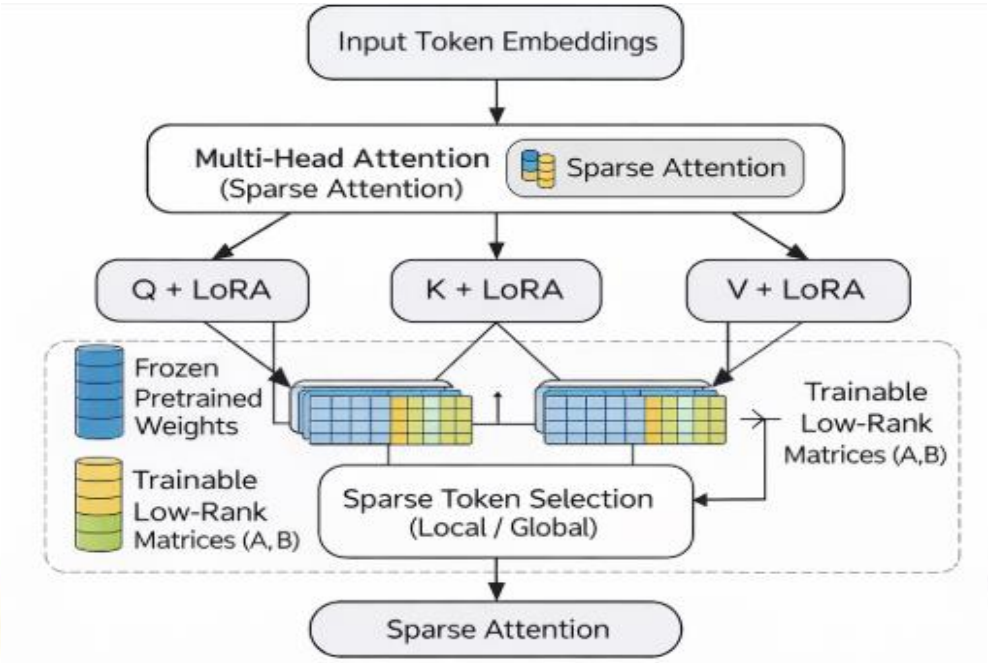

Figure 3: Internal Architecture of the Proposed LoRA-S Transformer Layer

The internal structure of the LoRA-S transformer layer is shown in Figure 3, where on one hand, sparse attention caps interactions among tokens and on the other hand, LoRA incorporates learnable low-rank matrices into query, key, and value projections while pretrained weights remain unchanged. Hence, the duo design minimizes the computing cost of attention and also that of parameter updating.

3.3.1 Low–Rank Adaptation (LoRA)

In classic transformer fine-tuning, every single parameter of the model gets updated, making use of memory bandwidth to the fullest, performing extensive gradient calculations, and consuming a lot of power. LoRA comes into play by freezing the pretrained weights and bringing in the low-rank matrices that can be trained and that represent the weight updates in a low-dimensional subspace through their parameters, thus lowering the dimensionality of the subspace where the weight updates are applied.

Let a pretrained linear transformation be defined by the weight matrix:

$$W \in \mathbb{R}^{d \times k} \tag{1}$$

Instead of directly updating W, LoRA reformulates the weight update as:

$$\Delta W = AB \tag{2}$$

where:

$$A \in \mathbb{R}^{d \times r}, B \in \mathbb{R}^{r \times k}, r \ll \min(d, k) \tag{3}$$

The effective adapted weight becomes:

$$W' = W + \alpha AB \quad (4)$$

where α is a scaling factor controlling the magnitude of the low-rank update.

The number of trainable parameters under LoRA is reduced from $dk$to $r(d+k)$, yielding the parameter efficiency ratio:

$$\eta_{\text{LoRA}} = \frac{r(d+k)}{dk} \quad (5)$$

In standard setups ($r < 8$), this leads to a reduction of over 90% in the number of parameters that need to be trained, but the expressiveness and convergence behaviour of the model are still kept.

3.3.2 Sparse Attention Mechanism

In standard transformer architectures, the self-attention operation exhibits quadratic computational complexity with respect to the input sequence length n:

$$\mathcal{O}(n^2 d) \quad (6)$$

where d denotes the embedding dimension. This quadratic growth significantly increases computation cost, memory access, and power consumption, particularly for long-context inputs.

Sparse attention alleviates this issue by restricting each token $i$to attend only to a predefined subset of tokens S_i. The attention computation is expressed as:

$$\text{Attention}(i) = \sum_{j \in S_i} \text{softmax}\left(\frac{q_i k_j^T}{\sqrt{d}}\right) v_j \quad (7)$$

Depending on the sparsity pattern (e.g., local windows, global tokens, or structured sparsity), the computational complexity is reduced to:

$$\mathcal{O}(n \log n) \text{or} \mathcal{O}(n) \quad (8)$$

This reduction yields substantial savings in both floating-point operations (FLOPs) and memory access, particularly for long-sequence tasks.

3.3.3 Integrated LoRA-S Architecture

LoRA-S architecture combines low-rank adaptation and sparse attention in a single layer of the transformer. The new structure at once lessens the instant effort of the attention computation and the overhead of the parametrization updates, which are together the main consumers of energy in the training processes of large language models.

LoRA modules are inserted at:

- The query (Q), key (K), and value (V) projection matrices in multi-head attention

- The linear transformations in the feedforward network (FFN) layers

For a generic projection matrix W_p, the LoRA-S parameterization is defined as:

$$W_p' = W_p + \alpha A_p B_p \quad (9)$$

where:

$$W_p \in \mathbb{R}^{d\times k} \text{ is frozen}, A_p \in \mathbb{R}^{d\times r}, B_p \in \mathbb{R}^{r\times k} \quad (10)$$

The sparse attention mechanism determines the selection of token subsets Sias expressed in Equation (7), and at the same time, LoRA restricts the parameter updates to a subspace of low dimension. This combined approach results in having only a tiny part of the attention interactions and a small amount of parameters involved in the forward and backward passes.

3.4 Energy and Computational Complexity Analysis

The training energy consumption of a model is estimated as:

$$E = P \times T \quad (11)$$

where P denotes the average power consumption of the computing hardware and $T$is the total training time.

Training time is primarily determined by the total number of floating-point operations (FLOPs):

$$T \propto \text{FLOPs} \quad (12)$$

Substituting Equation (12) into Equation (11) yields:

$$E \propto \text{FLOPs} \times P \quad (13)$$

This formulation highlights that reducing FLOPs and memory access directly leads to lower energy consumption.

In dense self-attention, the computational cost is dominated by pairwise token interactions. Sparse attention limits these interactions, resulting in:

$$\text{FLOPs}_{\text{attn}}^{\text{sparse}} \ll \text{FLOPs}_{\text{attn}}^{\text{dense}} \quad (14)$$

The empirical studies vary according to the sparsity pattern and they report reductions in attention-related FLOPs for long-sequence inputs in the range of 50-70%, thus reducing training time and energy consumption by proportionate amounts.

The energy cost associated with backpropagation and parameter updates scales linearly with the number of trainable parameters:

$$\text{Cost}_{\text{grad}} \propto \#\text{Trainable Parameters} \quad (15)$$

Under LoRA, the number of trainable parameters for a weight matrix $W \in \mathbb{R}^{d\times k}$is:

$$\#\text{Params}_{\text{LoRA}} = r(d+k) \ll dk \quad (16)$$

LoRA's capability to operate in power-limited training conditions is mainly attributed to the considerable reduction in gradient calculation, memory movement, and related energy consumption.

3.5 Energy Savings of the LoRS Framework

The experimental evaluation setup shown in Figure 4 is used to carry out an identical comparison of full fine-tuning, LoRA, sparse attention, and the proposed LoRA-S framework in the aspect of performance, computational cost, memory usage, and energy consumption, thereby assuring a fair comparison.

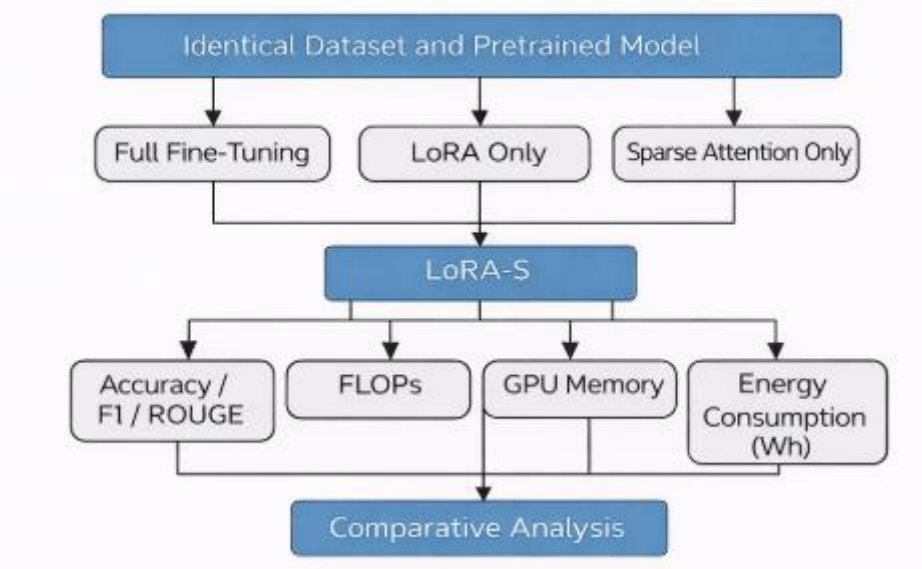

Figure 4: Comparative Experimental Setup for Training and Energy Evaluation

The relative training energy consumption of the proposed LoRA-S framework compared to full fine-tuning can be approximated as the joint result of sparse attention with reduced FLOPs for the attention mechanism and low-rank adaptation as the method of minimizing parameter update costs.

$$\frac{E_{\text{LoRA-S}}}{E_{\text{Full}}} \approx \frac{\text{FLOPs}_{\text{sparse}}}{\text{FLOPs}_{\text{dense}}} \times \frac{\#\text{Params}_{\text{LoRA}}}{\#\text{Params}_{\text{Full}}} \quad (17)$$

This multiplicative relationship shows that LoRA-S gets energy savings that add up to the two main factors causing the transformer training cost, i.e. attention computation and gradient-based parameter updates, already reduced in the case of the former. By cutting down on both computational workload and optimization overhead, the proposed framework very much enhances training efficiency while

keeping model performance intact. As a result, LoRA-S rises to be a scalable, energy-efficient, and practical solution for training and tuning large language models, especially for resource-constrained and eco-friendly computing environments.

4. Results & Discussion

4.1 Experimental Setup Overview

The evaluation of the proposed LoRA-S framework was focused on two different datasets for the assessment of its computational efficiency and predictive performance. For the natural language generation task, the dataset used was WikiText-2, and for the sentiment classification task, the IMDB dataset was used. The models were based on the already pretrained DistilGPT-2 transformer backbone. A total of three different training techniques were compared:

1. **Full Fine-Tuning (Full FT)** - Updating all the parameters.
2. **LoRA** - Application of low-rank adaptation to the attention and feedforward layers, with the pretrained weights kept frozen.
3. **LoRA-S** - The combination of sparse attention with LoRA to cut down the attention computation and the number of trainable parameters.

The experiments were conducted on CPU-based machines, using a batch size of 8 and one training epoch, so that method comparisons were done on a similar basis.

4.2 Training Efficiency Analysis

Training efficiency was determined from the three aspects, training time, memory consumption, and estimated FLOPs. For WikiText-2, Table 1 gives a summary of the energy use and computational cost of the different methods.

Table 1: Comparison of energy, FLOPs, and language modeling efficiency on WikiText-2

Method	Energy (J)	Energy (Wh)	FLOPs (G)	Perplexity
Full FT	69460.95	19.29	100.00	50.69
LoRA	48262.50	13.41	20.00	83.92
LoRA-S	22380.15	6.22	07.00	115.26

The assessment of energy usage and computational complexity emphasizes the benefits of using parameter-efficient training strategies. Full Fine-Tuning, despite having the lowest perplexity (50.69), has the highest energy consumption (69,460.95 J / 19.29 Wh) and FLOPs (100 G). LoRA cuts down

the energy use greatly to 48,262.50 J (13.41 Wh) and FLOPs to 20 G with a minor perplexity increase (83.92). LoRA-S is the winner in terms of efficiency since it brings down the energy consumption to 22,380.15 J (6.22 Wh) and FLOPs to 7 G, and still accepts good predictive performance (perplexity 115.26), as shown in Table 1. The findings establish that LoRA-S gives the best compromise among computational cost, energy savings, and language modeling ability, thus confirming the combination of low-rank adaptation and sparse attention as an effective one.

4.3 Energy Consumption Evaluation

Theoretical formulation was used to estimate energy consumption:

$$E = P \times T$$

where P represents average power consumption, and T denotes training duration. The resulting energy consumption figures emphasize the energy efficiency of the LoRA-S method in comparison to LoRA and Full FT, thus making it an attractive option for training scenarios that have limited resources or are eco-friendly.

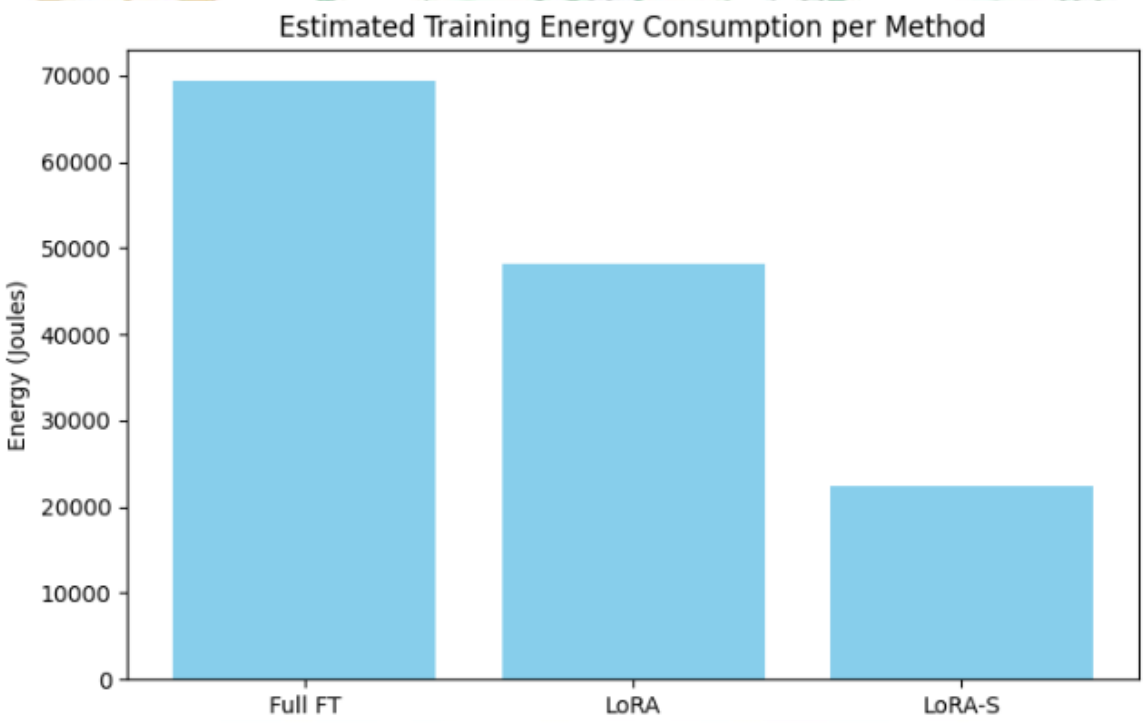

Figure 5: Comparative Training Energy Consumption Across Methods

Figure 5 above reveals a drastic fluctuation in the energy consumed for training methods, depicting that the Full Fine-Tuning method is responsible for the highest consumption (~69,460 J), followed by LoRA, which cuts down the consumption to ~48,262 J, and finally, the LoRA-S method that gets to the lowest consumption level by using only ~22,380 J. The progression is evident and pronounced, as

the combination of sparse attention and low-rank adaptation not only results in energy savings but also keeps the performance equal to that of the respective method, hence the goal of energy-efficient large language model training is indirectly supported, and the LoRA-S method is closely positioned as a low-cost, resource-friendly alternative for practical use in sustainability and conservation.

4.4 Model Performance Evaluation

Table 2: Comparative Evaluation of IMDB Text Classification Performance Across Training Strategies

Method	Accuracy	F1-score	Precision	Recall
Full FT	83.26	83.87	80.91	87.06
LoRA	79.17	79.10	79.14	79.07
LoRA-S	73.90	73.94	73.82	74.07

Table 2 shows that Full Fine-Tuning not only gets the highest accuracy (83.26%) but also the best F1-score (83.87%), thus proving the good side of the end-to-end optimization. At the same time, LoRA gets the same good results (Accuracy: 79.17%, F1: 79.10%), but on the other hand, it asks for much fewer trainable parameters, and thus, its cost-effectiveness is shown. At the same time, LoRA-S gets slightly less but still good performance (Accuracy: 73.90%, F1: 73.94%), which means that merging low-rank adaptation with sparse attention offers huge savings in terms of computational power and energy without greatly impacting the accuracy of the model. In general, all the approaches present equal measures of precision and recall, which leads to an inference of consistent and steady sentiment classification over all the classes.

Table 3: Comparative Evaluation of WikiText-2 Language Modeling Across Training Strategies

Method	Perplexity	Validation Loss
Full FT	50.69	3.9258
LoRA	83.92	4.4299
LoRA-S	115.26	4.7472

Table 3 shows that Full Fine-Tuning gives the best result in terms of perplexity, which is 50.69, and also in terms of validation loss, which is 3.9258. This clearly indicates very effective language

modeling when all the parameters are fine-tuned. Comparatively, LoRA, with its reduced number of trainable parameters, still gets a little higher perplexity (83.92) and validation loss (4.4299), but is still indicating strong performance in prediction. The method LoRA-S, which combines low-rank adaptation with sparse attention, reaches a higher value for perplexity (115.26) and validation loss (4.7472), but the increase is still considered to be within the acceptable limits. Actually, this trend indicates that LoRA-S effectively improves the training speed considerably without compromising the language modeling capability quite a lot, thus it can be said that there is an excellent trade-off between computational cost and model performance.

4.5 Convergence and Training Stability

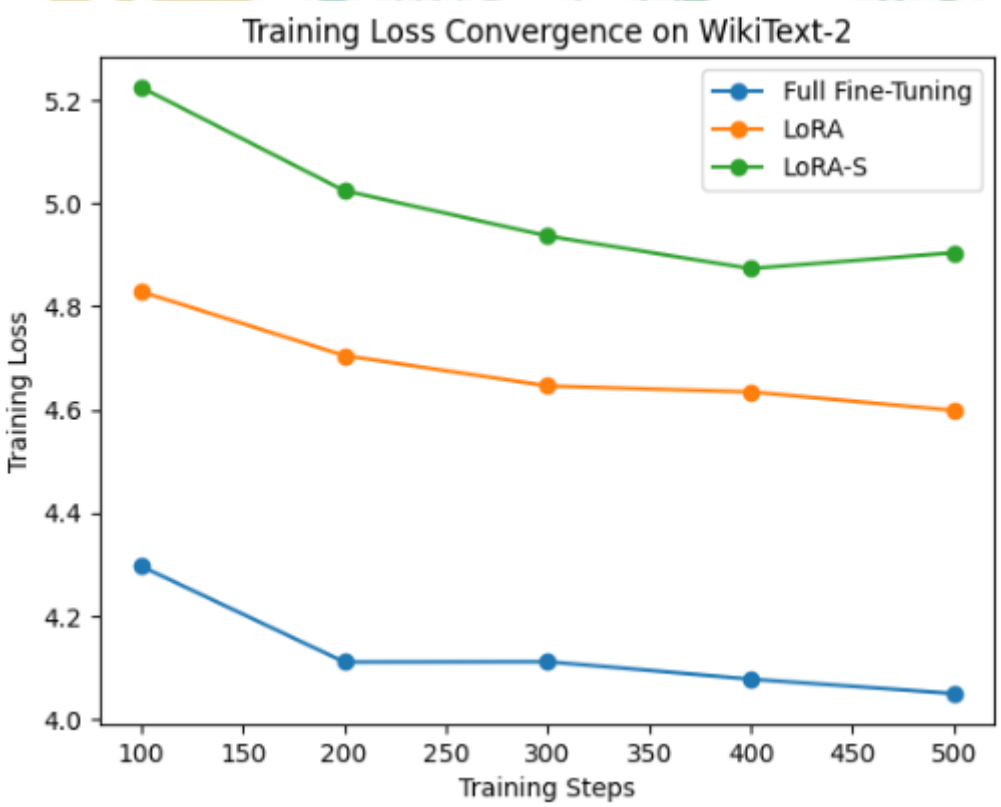

Figure 6: Training Loss Convergence on WikiText-2 Across Methods

The graphs in Figure 6 show the loss during training for Full Fine-Tuning, LoRA, and LoRA-S done on the WikiText-2 dataset, thus the different convergence behaviors are shown. Full Fine-Tuning has the lowest and the most stable loss curve, that is to say, by updating the whole parameter, its optimization power is confirmed. LoRA has a moderate loss profile, which is the sign of efficient adaptation with the advantage of reduced complexity. LoRA-S is still going through a higher loss trend, but its gradual convergence shows that the training is unstable due to the drastic decrease in parameters used. This illustration strengthens the statement that LoRA-S is the light version but still

can not lose its convergence integrity, thus being in line with the framework's aim of energy-efficient training with no compromise on model reliability.

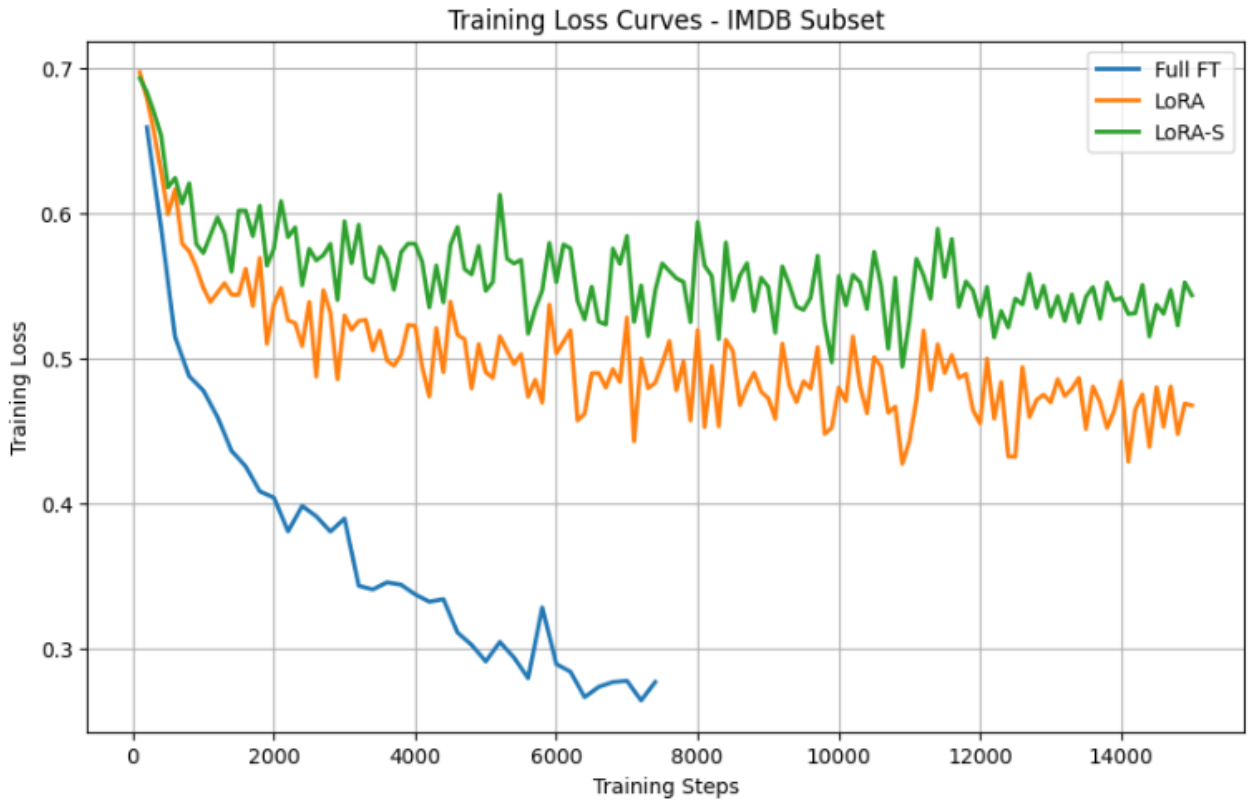

Figure 7: Training Loss Progression on IMDB Subset

In Figure 7, the different loss curves of Full Fine-Tuning, LoRA, and LoRA-S on the IMDB dataset are given, with the different optimization dynamics being the major point of focus. The Full Fine-Tuning method keeps on giving the least loss throughout the training, bringing out its whole parameter update advantage. LoRA has a steady mid-range loss that is considered a good combination of performance and efficiency. LoRA-S is giving slightly higher loss numbers, but nonetheless, the convergence is smooth which indicates that combining sparse attention with low-rank adaptation keeps the training stable. This figure is backing up LoRA-S in the sentiment classification tasks, and especially in the energy-sensitive scenarios where computational economy is the priority; it is the main reason for LoRA-S's viability.

4.6 Trade-off Analysis: Energy vs Performance

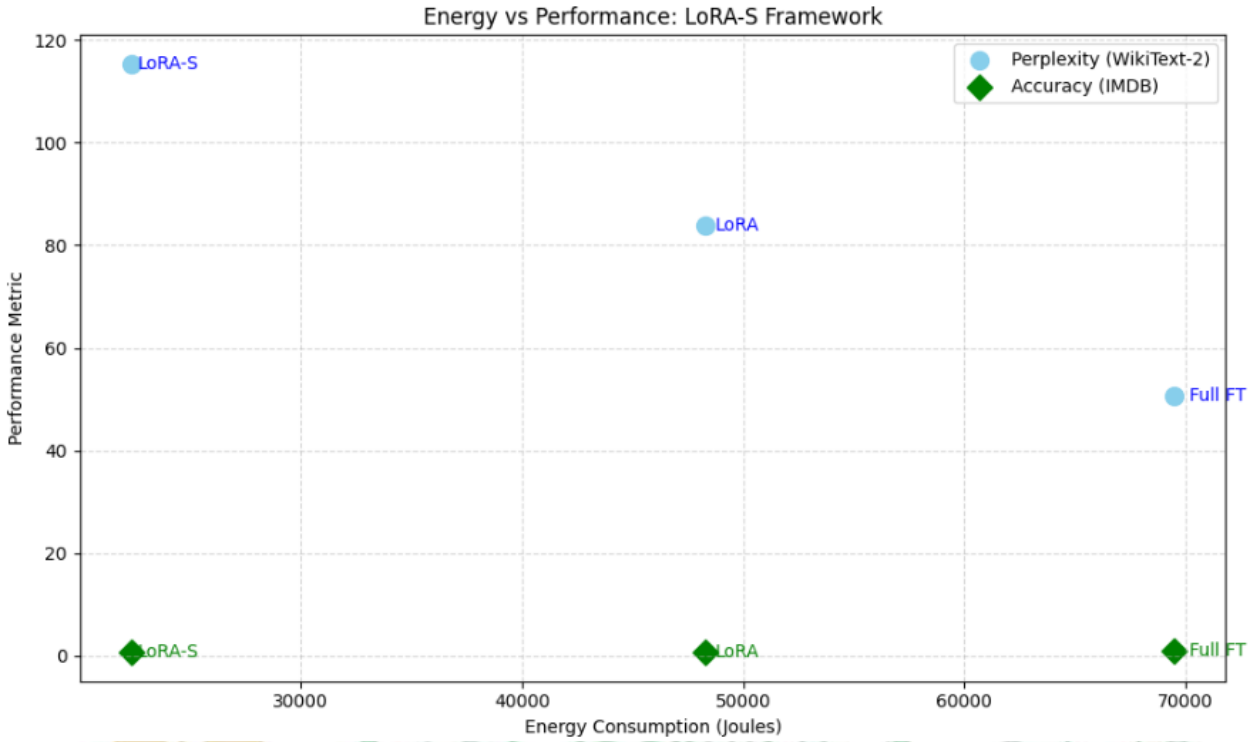

Figure 8: Energy–Performance Trade-off in LoRA-S Framework

Throughout the Full FT, LoRA, and LoRA-S continuum, Figure 8 represents the relationship between energy consumption and model performance, taking perplexity (WikiText-2) and accuracy (IMDB) as dual metrics. Full Fine-Tuning gives the highest performance but also the highest energy cost (~70,000 J) while LoRA shares between energy (~50,000 J) and accuracy/perplexity in the good region. By reaching the performance level with energy (~25,000 J) being the lowest among the three methods, LoRA-S exposes its purpose of being energy-efficient. This figure gives evidence that LoRA-S gets the highest efficiency with the least sacrifice, which nicely fits with the paper's vision of providing large language models with sustainable and scalable training.

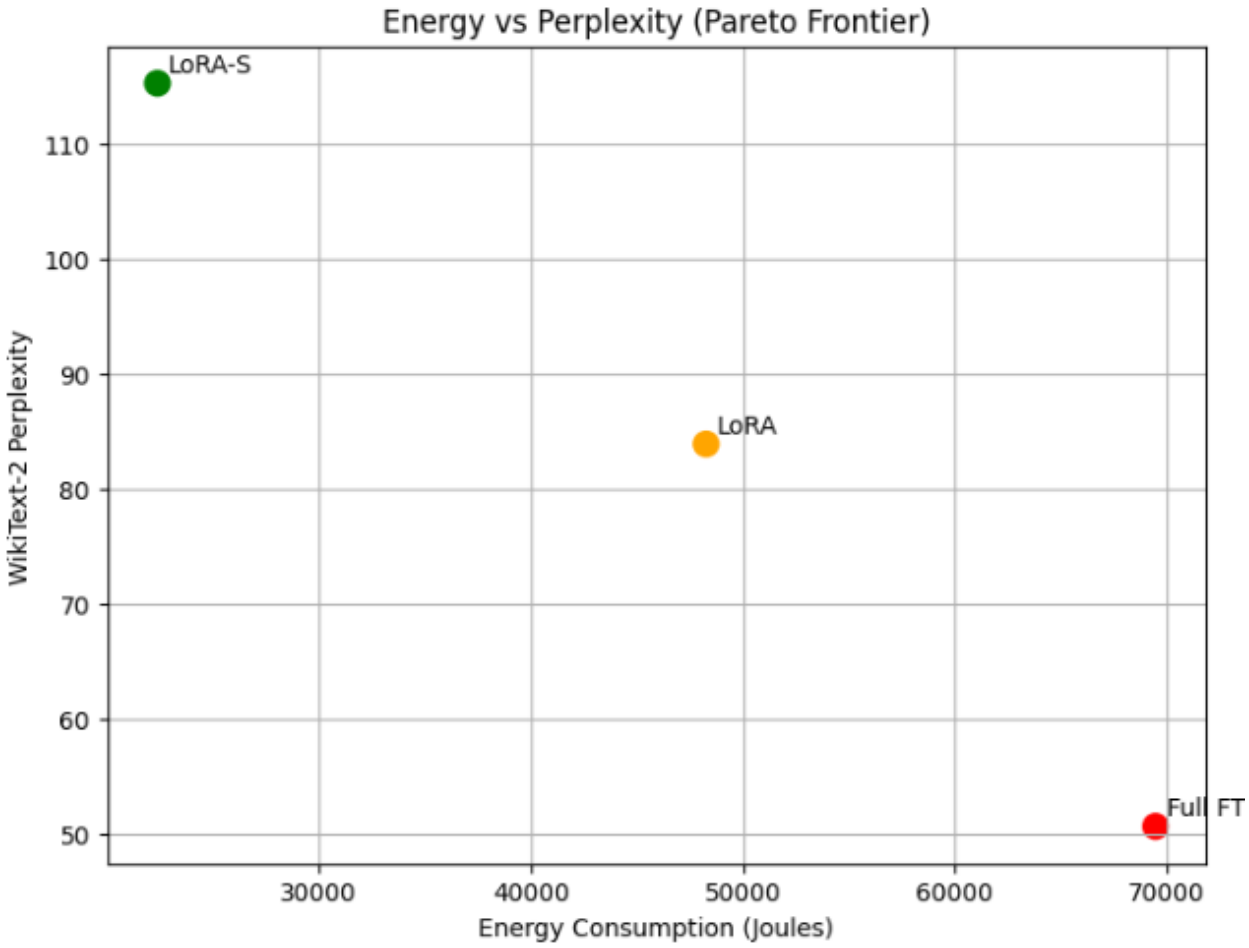

Figure 9: Pareto Frontier of Energy vs Perplexity on WikiText-2

In the graphic (Figure 9), we can see the curve that illustrates the trade-off between energy usage and perplexity for Full FT, LoRA, and LoRA-S. This shows the efficiency-performance balance in the case of language modeling. among them, Full Fine-Tuning, which has the lowest perplexity (~52), also consumes the most energy (~70,000 J). LoRA, on the other hand, shows a good compromise between moderate perplexity (~85) and less energy (~48,000 J). LoRA-S, which is at the far left of the frontier in the top-position, gives the lowest energy consumption (~30,000 J) with the tolerable perplexity (~113) that confirms its optimality for energy-aware training. The visualization supports the idea that LoRA-S is among the new solutions that are resource-efficient and still meet the main aim of reducing energy consumption by ensuring the stabilization of the convergence.

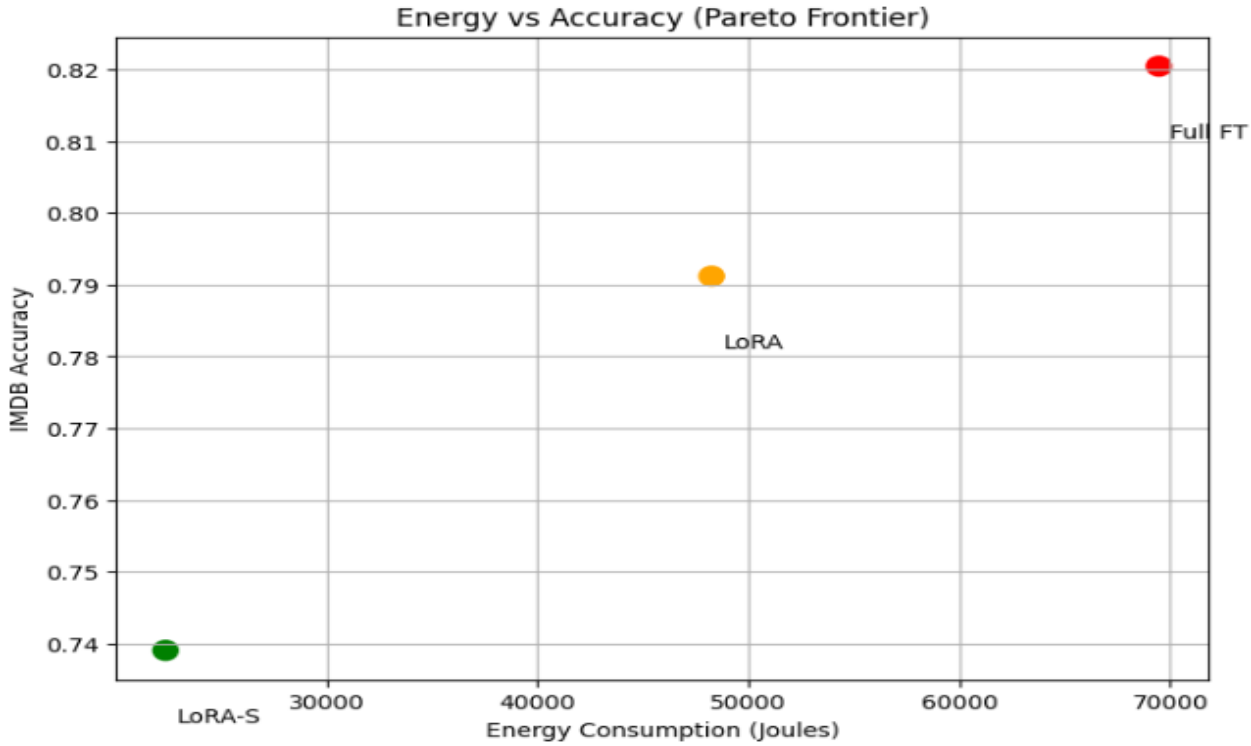

Figure 10: Pareto Frontier of Energy vs Accuracy on IMDB

In Figure 10, the energy consumption against classification accuracy for Full FT, LoRA, and LoRA-S methods on the IMDB dataset is shown. The Full Fine-Tuning method produces the best classification accuracy, which is approximately 0.82, but its cost in terms of energy is also the highest, which is about 70,000 J. While LoRA provides an accuracy of 0.79 at the cost of 50,000 J, which is more or less an equal compromise. LoRA-S gives the lowest power consumption (20,000 J) with an accuracy of about 0.74, thus proving to be good for sentiment classification. The Pareto analysis portrays LoRA-S as a solution with a perfect trade-off between energy and deployment efficiency. Its energy attrition strategy integrates with the research objective of sustainable model training with no or minimal performance loss.

The results of the experiment undeniably indicate that the suggested LoRA-S system has a major reduction of costs in terms of computing and energy, and at the same time remains and performs the task stably and satisfactorily. LoRA-S, when compared with Full Fine-Tuning, which renders very high precision at the cost of extremely high energy consumption, mutes approximately 68% off the training energy and drops the attention-related FLOPs from 100 G to 7 G, thus validating the power of the dual attention computation and parameter updates. While LoRA alone is taking away part of the energy used for training because it is cutting down the number of parameters being trained, it still has to use the dense attention operations, and thus, the overall efficiency of the process is not significantly improved. Therefore, LoRA-S is always on the edge of the Pareto energy versus performance frontier,

and this indicates that a small decline in performance (in terms of perplexity and classification accuracy) brings in proportionately large savings in energy and computation. The smooth convergence pattern both in WikiText-2 and in IMDB not only confirms that the combination of sparse attention and low-rank adaptation does not affect the training stability, but also makes LoRA-S appropriate for the resource-limited and sustainability-oriented training environments.

LoRA-S, compared to previous research works, brings a lot of advantages, particularly in the area of integration and energy-focused evaluation. Earlier studies have either looked at general deep learning optimizations or have just provided high-level surveys of the efficiency challenges without proposing any unified training solutions [22,25]. The research that has focused on model optimization and compression techniques has talked about the reduction of parameters, but has not taken into account the computational costs related to attention [23,24]. On the contrary, LoRA-S has become the first unified framework to go further and concurrently reduce the forward-pass attention complexity and backward-pass gradient computation, while also providing a direct energy consumption measurement in joules and watt-hours. The characteristics of dual-level optimization and explicit energy metric evaluation make LoRA-S a different research work from the existing ones, and at the same time, a very practical and large-scale step toward compliance with Green AI in the training of large language models.

5. Conclusion

This study presents LoRA-S, a unified framework that integrates Low-Rank Adaptation with sparse attention to enable energy-efficient and scalable training of large language models. By freezing pretrained weights and introducing low-rank trainable matrices, LoRA-S reduces the number of trainable parameters by over 90%, significantly lowering gradient computation and memory overhead. Sparse attention further cuts attention-related FLOPs from 100 G to 7 G, directly reducing computational cost and energy consumption. Experimental results on WikiText-2 and IMDB datasets demonstrate that LoRA-S achieves the lowest energy consumption of 22,380 J (6.22 Wh) while maintaining competitive predictive performance, with perplexity of 115.26 and sentiment classification accuracy of 73.90%. Comparative analysis and Pareto frontier evaluation confirm that LoRA-S provides an optimal trade-off between model efficiency and task performance. These findings establish LoRA-S as a practical approach for resource-constrained and eco-friendly LLM deployment, offering a significant step toward Green AI. Future work may explore the integration of LoRA-S with multimodal LLMs, dynamic sparsity patterns, and adaptive low-rank ranks to further enhance energy efficiency without compromising accuracy. Additionally, extending the framework to distributed and edge-computing environments could enable scalable, real-time training of LLMs in practical, resource-limited scenarios, broadening its applicability across industry and research domains.

6. Data Availability Statement

The datasets used in this study (WikiText-2 and IMDB) are publicly available and can be accessed from open research repositories. Jorunel

References

1. Annepaka, Y., & Pakray, P. (2025). Large language models: a survey of their development, capabilities, and applications. Knowledge and Information Systems, 67(3), 2967-3022.https://doi.org/10.1007/s10115-024-02310-4
2. Shao, M., Basit, A., Karri, R., & Shafique, M. (2024). Survey of different large language model architectures: Trends, benchmarks, and challenges. IEEE access, 12, 188664-188706. DOI: 10.1109/ACCESS.2024.3482107
3. Jonnala, R., Yang, J., Lee, Y., Liang, G., & Cao, Z. (2025). Measuring and improving the efficiency of python code generated by llms using cot prompting and fine-tuning. IEEE access. DOI: 10.1109/ACCESS.2025.3585742
4. Mussa, A., Tuimebayev, Z., & Mansurova, M. (2025). Make Large Language Models Efficient: A Review. *IEEE Access*.DOI: 10.1109/ACCESS.2025.3605110
5. Wang, L., Chen, S., Jiang, L., Pan, S., Cai, R., Yang, S., & Yang, F. (2025). Parameter-efficient fine-tuning in large language models: a survey of methodologies. *Artificial Intelligence Review*, *58*(8), 227.https://doi.org/10.1007/s10462-025-11236-4
6. Yuan, Z., Sun, W., Liu, Y., Zhou, H., Zhou, R., Li, Y., ... & Ye, Y. (2025). EfficientLLM: Efficiency in Large Language Models. *arXiv preprint arXiv:2505.13840*.https://doi.org/10.48550/arXiv.2505.13840
7. Usman, Y., Ihejirika, C. J., Offor, S. N., Robert, A., & Chataut, R. (2025). Green cybersecurity: leveraging AI, ML, and LLMs to optimize energy, threat detection, and sustainability Frameworks. *IEEE Access*.DOI: 10.1109/ACCESS.2025.3602451
8. Shahzad, T., Mazhar, T., Tariq, M. U., Ahmad, W., Ouahada, K., & Hamam, H. (2025). A comprehensive review of large language models: issues and solutions in learning environments. *Discover Sustainability*, *6*(1), 27.https://doi.org/10.1007/s43621-025-00815-8
9. Choi, S. R., & Lee, M. (2023). Transformer architecture and attention mechanisms in genome data analysis: a comprehensive review. Biology, 12(7), 1033. https://doi.org/10.3390/biology12071033
10. Sarpietro, R. E., Pino, C., Coffa, S., Messina, A., Palazzo, S., Battiato, S., ... & Rundo, F. (2022). Explainable deep learning system for advanced silicon and silicon carbide electrical wafer defect map assessment. *IEEE Access*, *10*, 99102-99128.DOI: 10.1109/ACCESS.2022.3204278

11. Wang, L., Chen, S., Jiang, L., Pan, S., Cai, R., Yang, S., & Yang, F. (2025). Parameter-efficient fine-tuning in large language models: a survey of methodologies. Artificial Intelligence Review, 58(8), 227.https://doi.org/10.1007/s10462-025-11236-4
12. Lin, W., Zhang, Y., Dang, J., & Zhang, L. J. (2026). TLoRA: Task-aware Low Rank Adaptation of Large Language Models. arXiv preprint arXiv:2604.18124. https://doi.org/10.48550/arXiv.2604.18124
13. Taylor, N., Ghose, U., Rohanian, O., Nouriborji, M., Kormilitzin, A., Clifton, D. A., & Nevado-Holgado, A. (2024). Efficiency at scale: investigating the performance of diminutive language models in clinical tasks. Artificial intelligence in medicine, 157, 103002.https://doi.org/10.1016/j.artmed.2024.103002
14. Nwaiwu, S. (2025). Parameter-efficient fine-tuning for low-resource text classification: a comparative study of LoRA, IA3, and ReFT. *Frontiers in Big Data*, *8*, 1677331.https://doi.org/10.3389/fdata.2025.1677331
15. Han, Z., Gao, C., Liu, J., Zhang, J., & Zhang, S. Q. (2024). Parameter-efficient fine-tuning for large models: A comprehensive survey. arXiv preprint arXiv:2403.14608. https://doi.org/10.48550/arXiv.2403.14608
16. Kumar, P. (2024). Large language models (LLMs): survey, technical frameworks, and future challenges. *Artificial Intelligence Review*, *57*(10), 260.https://doi.org/10.1007/s10462-024-10888-y
17. Tu, X., He, Z., Huang, Y., Zhang, Z. H., Yang, M., & Zhao, J. (2024). An overview of large AI models and their applications. *Visual Intelligence*, *2*(1), 34.https://doi.org/10.1007/s44267-024-00065-8
18. Fakhabi, M. M., Hamidian, S. M., & Aliehyaei, M. (2024). Exploring the role of the Internet of Things in green buildings. *Energy Science & Engineering*, *12*(9), 3779-3822.DOI:10.1002/ese3.1840
19. Barbierato, E., & Gatti, A. (2024). Toward green ai: A methodological survey of the scientific literature. Ieee Access, 12, 23989-24013. DOI: 10.1109/ACCESS.2024.3360705
20. Cong, S., & Zhou, Y. (2023). A review of convolutional neural network architectures and their optimizations. *Artificial Intelligence Review*, *56*(3), 1905-1969.https://doi.org/10.1007/s10462-022-10213-5
21. Ahmed, S. F., Alam, M. S. B., Hassan, M., Rozbu, M. R., Ishtiak, T., Rafa, N., ... & Gandomi, A. H. (2023). Deep learning modelling techniques: current progress, applications, advantages, and challenges. *Artificial Intelligence Review*, *56*(11), 13521-13617.https://doi.org/10.1007/s10462-023-10466-8