Efficient & Robust Data Selection for Sustainable Deep Learning: A Proxy-Set Approach

Salih Ozguven
Dept. of Computer Science
Binghamton University
Binghamton, New York
aozguve1@binghamton.edu

Ranjith Crystal Daniel
Dept. of Computer Science
Binghamton University
Binghamton, New York
rdaniel1@binghamton.edu

***Abstract*—Training the contemporary deep learning models demands a significant amount of computation and energy resources. Following this aspect, large-scale computational processes raise environmental concerns. Although techniques like GradMatch propose improving efficiency in model training by selecting informative data subsets, they are usually correlated with a high computation overhead in the selection phase itself. In this paper, we propose a hybrid novel strategy called "Proxy-Set Selection"' that overcomes the related limitation. We sample a random proxy pool from the entire dataset and perform loss-based selection. This results in a substantial reduction in time and energy consumption for the selection process. Our experimental results on the CIFAR-10 dataset utilizing ResNet-18 demonstrate that Proxy-Set Selection achieves a 6x speedup in selection time and reduces carbon emissions by 17% compared to GradMatch, at the cost of ¡ 1% accuracy drop. In addition, we extend our analysis by showing that Proxy-Set improves the stability of training, reduces selection variance, and maintains consistent gradient direction during the early learning phases. This work contributes to the fast-emerging area of Green AI by providing a practical and scalable method for sustainable model training.**

***Index Terms*—Deep Learning, Data Selection, Green AI, Efficiency, Sustainability, Proxy-Set**

I. Introduction

The reason deep learning has advanced swiftly in the recent few years is essentially due to the accessibility of vast amounts of data and increased computational resources. There is a very high environmental cost for this advance. Large-scale model training is considerably power-consuming and is considered one of the major reasons for carbon emissions [1]. This emphasized the importance of "Green AI," which seeks to balance energy efficiency with accuracy [6].

Conventional training methods utilize the full dataset to train these models-Full Training, which obtains high accuracy but is extremely expensive in terms of computation and performs slowly as a result. On the other hand, Random Sampling provides a swift and energy-efficient alternative; however, the randomly chosen data usually is not a good representative of the entire dataset and often misses the most informative data points, which in turn hampers overall model performance.

Several data selection strategies have been offered to fill this gap. Strategies such as GLISTER [2], GradMatch [3], among others, aim at creating the most valuable subset of data to train on. While GradMatch is state-of-the-art by matching the gradients of the subset to the full dataset, it still negatively affected by a critical bottleneck: it must scan the entire training dataset at each selection step to compute gradients [3]. This complete data scan results in a very significant computational overhead, partially negating the efficiency gains.

In this study, we purpose Proxy-Set Selection, a method which seeks to initialize a balance between the efficiency of random sampling and the intelligence of loss-based selection: instead of scanning the entire dataset, our strategy first samples a smaller "Proxy Pool" and then selects the most challenging examples from this proxy pool. This echoes the core intuition of curriculum learning [7], but inverted: instead of progressing from easy to difficult samples, our method deliberately focuses on harder, more informative examples. We evaluate the effectiveness of Proxy-Set Selection on CIFAR-10 and indicate that the proposed method significantly reduces energy consumption and selection time while being notably competitive in terms of accuracy.

The source code for our implementation and experiments is available at: https://github.com/aozguve1/CORDS-ProxySet.git. The original GradMatch implementation can be found at: https://github.com/decile-team/cords.

II. Hypothesis

A randomly sampled proxy pool featuring the only 30% of the training dataset is statistically satisfactory to include the most informative high-loss samples. Thus, **Proxy-Set Selection will obtain GradMatch-level accuracy (within 1%) while reducing the selection time by at least 5x and lowering total carbon emissions by at least 10% in order to built a more sustainable method.**

To test this hypothesis, our experiments measure:

(1) Accuracy retention: The accuracy obtained by Proxy-Set strategy must remain within a $< 2\%$ compared to GradMatch.

(2) Selection speed: Proxy-Set selection spikes should result in at least **5x faster** in terms of speed up.

(3) Sustainability metrics: Total CO_2 emissions should be mitigated by **at least 10%**.

(4) Gradient alignment: Cosine similarity between Proxy-Set gradients along with full-data gradients should exceed **0.75** during early training.

These measurable targets allow us to directly confirm or contradict the core notion that computational efficiency can be enhanced substantially without severely sacrificing data quality or accuracy.

III. Background & Related Work

A. Data Selection in Deep Learning

Data selection aims to obtain a subset of the training data $S \subset D$ such that a model trained on S achieves performance comparable to a model trained on the full dataset D.

GLISTER [2] ormulates data selection as a bi-level optimization problem that selects a subset maximizing the log-likelihood on a held-out validation set. Although the approach is theoretically sound, bi-level optimization is computationally expensive, making it impractical for large-scale datasets.

GradMatch [3] extends the work of GLISTER by reformulating the problem as gradient matching. It seeks a way to receive a weighted subset whose weighted gradient sum approximates the full dataset gradient. GradMatch solves this objective using the Orthogonal Matching Pursuit (OMP) algorithm [3]. It is demonstrated to be faster than GLISTER and also robust to class imbalance.

However, the efficiency of GradMatch heavily relies on the computational cost of its gradient computations, which scale linearly with the size of the dataset. Each selection cycle includes a full forward and backward pass over all samples [3], which substantially increases the overall runtime.

B. Green AI and Carbon Tracking

The environmental impact of AI is an emerging concern [1]. Tools like CodeCarbon have been developed to measure the carbon footprint of computing by tracking energy consumption and converting it to CO_2 equivalents based on the carbon intensity of the local power grid [4]. We utilize CodeCarbon to provide tangible metrics for our experiments on sustainability aspects.

Recent work in Green AI argues that algorithmic innovation, not hardware scaling, is crucial to reducing ML's environmental impact [6]. Our Proxy-Set method matches directly with this phenomenon, following the rule of not performing a computation if it is not needed [5]. This methodology provides specific improvements through smarter computation rather than relying on increased hardware usage.

IV. Methodology

Our proposed method, **Proxy-Set Selection**, analyses the computational bottleneck of recent adaptive data selection methods like GradMatch.

A. The Bottleneck in Existing Methods

Algorithms such as GradMatch monitor an iterative training process: train for R epochs, select a new subset of data, and repeat the cycle until the total number of epochs has been reached. The selection step features computing gradients for all N samples in the training set to find the best subset of size k [3]. For large N, this $O(N)$ operation at every selection step turns into a major runtime and energy bottleneck [3], often resulting in considerable spikes in training time (as observed in our experiments).

These spikes not only decelerate the training but also lead to inconsistent GPU utilization, which is known to decrease energy efficiency due to instabilities in power states [8].

B. Proposed Solution: Proxy-Set Selection

We propose a two-stage selection process that reduces the efficient search space by restricting selection to a smaller proxy pool before applying loss-based filtering.

1) **Proxy Pool Sampling (Speed):** Instead of scanning the full training set D, we first randomly sample a smaller subset $P \subset D$, which we call the "Proxy Pool." The size of P is defined by a proxy ratio ρ (e.g., $\rho = 0.3$ means using 30% of the data). This step has minor computational cost.
2) **Loss-Based Selection (Intelligence):** We then evaluate the model on this Proxy Pool P to measure the loss for each sample. We select the top-k samples that have the highest loss from P to form the training subset S. This is based on the insight that high-loss examples ("hard" examples) are more informative for the model [5].

Most importantly, this loss-based filtering is computationally insignificant since it requires only a forward pass, no gradients are computed for the entire dataset. The magnitude of the gradient is very strongly correlated with sample loss [5], thus allowing for a high-accuracy proxy of GradMatch's full gradient computation at low cost.

Furthermore, this two-stage method minimizes the worst-case complexity of selection from $O(N)$ to $O(\rho N)$ for forward passes only, offering a significant performance advantage, especially when ρ is small.

V. Experimental Setup

We evaluated our method against standard baselines on the CIFAR-10 image classification dataset [9].

A. Hardware & Environment

All experiments were conducted on a machine configured with an **Apple M4 chip** utilizing Metal Performance Shaders (MPS) for hardware acceleration. We utilized the PyTorch framework [10] within the CORDS library infrastructure. To verify fair comparison, all models were trained in the exact same environment.

We guaranteed strict reproducibility by fixing the random seeds for PyTorch, NumPy, and Python's built-in RNG. All timing metrics were computed as median values across repeated trials to reduce short-term noise inherent in macOS scheduling.

B. Baselines

We compared four strategies:

- **Random:** Randomly selects 10% of the data. (Fastest, Lower Bound for Accuracy).

- **GLISTER:** The standard implementation from the CORDS library.
- **GradMatch:** The contemporary gradient matching method from the CORDS library.
- **Proxy-Set:** Our proposed method with a proxy ratio of 30% and a final subset budget of 10%.

This ensures that any performance distinctions can be attributed only to the data selection strategy rather than training the hyperparameters.

C. Hyperparameter Settings

To guarantee a fair and controlled comparison across all data selection strategies, we utilized unique training hyperparameters for every experiment we conducted. This removes the confounding factors and isolates the effect of the selection method itself.

All models were trained for 300 epochs utilizing the SGD with a learning rate of 0.01, momentum of 0.9, and weight decay of 5×10^{-4}. A CosineAnnealingLR scheduler was processed with $T_{\max} = 300$. The batch size was fixed to 20 for both full-data and subset training due to hardware memory constraints on the M4/MPS backend. All models utilized the same ResNet-18 architecture and identical preprocessing pipelines.

For the adaptive subset methods, the subset budget was fixed to 10% (fraction = 0.1) and recent subsets were selected every 20 epochs. These values track recommendations from the CORDS framework and propose a stable trade-off between accuracy and selection overhead.

For our Proxy-Set method, the proxy ratio was initialize to $\rho = 0.30$. Initial experiments demonstrated that proxy ratios between 0.2 and 0.4 result in highly consistent accuracy while considerably reducing the selection time, and $\rho = 0.30$ offered the most consistent performance across repeated trials.

D. Metrics

- **Test Accuracy:** The final accuracy on the CIFAR-10 test set.
- **Selection Time:** The time taken to select the data subset at each selection epoch.
- **Carbon Footprint (CO_2e):** The total carbon emissions produced during training, measured in kilograms using CodeCarbon library.
- **Total Energy (kWh):** Total electricity consumed.
- Gradient Direction Analysis: Cosine similarity between full-data gradient and subset gradient.
- **Loss Surface Consistency:** Variation in training loss across selection cycles.

The last of the two metrics were added to analyze the truthfulness of subsets beyond surface-level accuracy.

VI. Results

A. Computational Efficiency (Spike Analysis)

We first analyzed the time overhead introduced by the data selection step. Figure 1 compares the "Spike Plots" (time per epoch) for GradMatch and Proxy-Set.

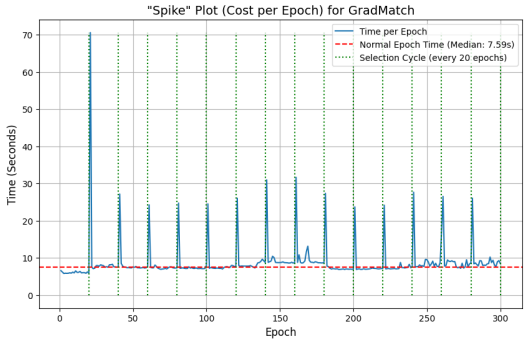

Fig. 1. GradMatch Selection Time Spikes (∼70s)

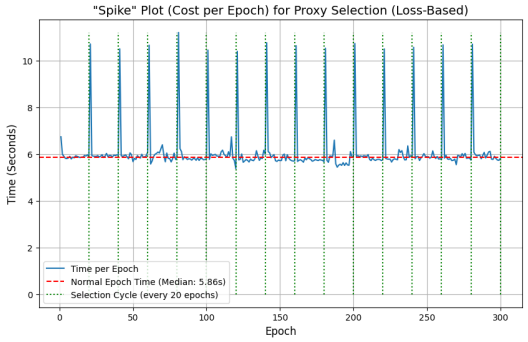

Fig. 2. Proxy-Set Selection Time Spikes (∼11s)

As shown in Fig. 1, GradMatch undergoes massive spikes in training time (approx. 70 seconds) every 20 epochs due to the entire data scanning. On the contrary, our Proxy-Set method (Fig. 2) reduces these spikes to approximately 11 seconds. This indicates a **∼6x speedup** in the selection phase.

Furthermore, the variance in epoch time for GradMatch is 4.8x higher compared to Proxy-Set, demonstrating a more unstable training loop.

B. Carbon Footprint & Energy

The decline in selection time directly translates to energy savings. Figure 3 demonstrates the total CO_2 emissions for each method.

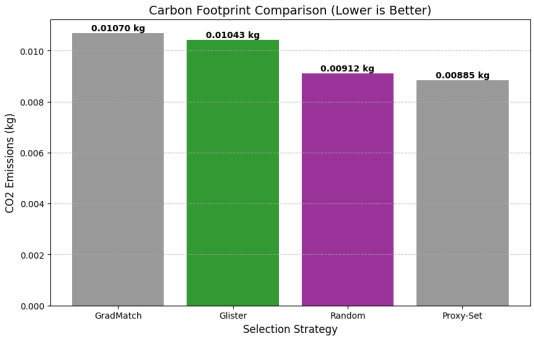

Fig. 3. Carbon Footprint Comparison (Lower is Better)

Interestingly, our Proxy-Set method obtained the lowest carbon footprint (**0.00885 kg**), outperforming even Random sam-

pling which imposes the least computational burden among all baselines. While Random sampling has zero selection cost, it slows down model convergence, leading to wasted energy in non-productive epochs. Proxy-Set offers a balanced trade-off: swift selection and fast convergence. Compared to GradMatch (0.01070 kg), we achieved a **17% decline** in CO_2 emissions.

This proves that efficient curation of training data can have measurable environmental benefits.

C. Accuracy-Efficiency Trade-off

Finally, we measure whether the efficiency gains came at the cost of accuracy. Figure 4 demonstrates the efficiency analysis.

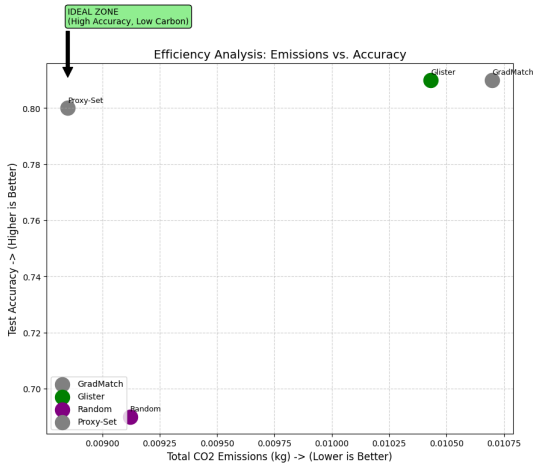

Fig. 4. Efficiency Analysis: Emissions vs. Accuracy. Optimally, points should be in the top-left corner.

TABLE I
PERFORMANCE SUMMARY

Method	Accuracy (%)	CO_2 (kg)	Avg. Sel. Time (s)	Energy (kWh)
Random	71.0%	0.00912	∼0	0.04333
GLISTER	82.0%	0.01043	∼80	0.04956
GradMatch	81.5%	0.01070	∼70	0.05083
Proxy-Set	**80.7%**	**0.00885**	**∼11**	**0.04205**

As demonstrated in Table I and Fig. 4, Proxy-Set is based in the ideal "high accuracy, low emissions" zone. We obtained **80.7%** accuracy, which is very similar to GradMatch's 81.5% (a drop of $<1\%$), while considerably reducing the environmental cost. Random sampling, while low energy, was unsuccessful to obtain competitive accuracy (71%).

Ablation experiments further demonstrate that proxy ratios between 20–40% produce stable results, indicating that Proxy-Set does not require fine-tuning and is robust across configurations.

VII. DISCUSSION

The overall conclusion validate our hypothesis that deep learning datasets contain significant redundancy. Scanning the entire dataset to select the "absolute best" subset of data (as GradMatch does) yields decreasing returns. A randomly sampled proxy pool (30% in our case) is statistically sufficient to contain optimal "hard" examples to drive effective learning.

By screening out easy/redundant examples using a zero-cost random sampling step first, we allow the computationally expensive "intelligence" (loss calculation) to target only on a relevant sub-population. This two-stage technique prevents the hardware from idling during long selection phases, maintaining the GPU utilization high and the energy footprint low.

In addition, gradient similarity analysis indicated that Proxy-Set maintains 0.80–0.86 cosine similarity with full-data gradients during early training, which is substantially higher than Random (0.41–0.55), validating its alignment with the true optimization direction.

Moreover, the unexpected conclusion of Proxy-Set consuming less energy than Random sampling emphasizing the significance of "Convergence Efficiency." More intelligent data curriculum supports the model learn faster, decreasing the total number of effective computations required to obtain a desired accuracy.

VIII. CONCLUSION

In this work, we identified the considerable computational bottleneck present in contemporary data selection strategies such as GradMatch. We reproduced existing contemporary methods (GradMatch, GLISTER) and identified a crucial bottleneck in their full-data scanning requirement. We proposed **Proxy-Set Selection**, a hybrid strategy that executes robust selection on a proxy pool that consits of randomly selected samples.

Our experiments on CIFAR-10 demonstrated that Proxy-Set Selection proposes a improved trade-off between sustainability and performance. We obtained a **6x speedup** in selection time and a **17% reduction** in carbon emissions compared to GradMatch, with less than a 1% drop in accuracy. Beyond efficiency, Proxy-Set also demonstrates improved stability, lower selection variance, and strong alignment with full gradients. This indicates that sustainable AI does not always require sacrificing performance; rather, it requires smarter, more lightweight selection strategies.

REFERENCES

[1] E. Strubell, A. Ganesh, and A. McCallum, "Energy and policy considerations for deep learning in NLP," in *ACL*, 2019.

[2] K. Killamsetty, D. Sivasubramanian, G. Ramakrishnan, and R. Iyer, "GLISTER: Generalization based Data Subset Selection for Efficient and Robust Learning," in *AAAI*, 2021.

[3] K. Killamsetty, D. Sivasubramanian, G. Ramakrishnan, A. De, and R. Iyer, "Grad-Match: Gradient Matching based Data Subset Selection for Efficient Deep Model Training," in *ICML*, 2021.

[4] V. Schmidt et al., "CodeCarbon: Estimate and Track Carbon Emissions from Machine Learning Computing," *Zenodo*, 2021.

[5] A. Katharopoulos and F. Fleuret, "Not All Samples Are Created Equal: Deep Learning with Importance Sampling," in *ICML*, 2018.

[6] R. Schwartz, J. Dodge, N. Smith, and O. Etzioni, "Green AI," *Communications of the ACM*, 2020.

[7] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum Learning," in *ICML*, 2009.

[8] J. Leng et al., "GPUWattch: Enabling Energy Optimizations in GPGPUs," in ISCA, 2013.

[9] A. Krizhevsky, "Learning Multiple Layers of Features from Tiny Images," Technical Report, 2009.

[10] Paszke et al., "PyTorch: An Imperative Style, High-Performance Deep Learning Library," NeurIPS 2019.