Proceedings of the 5th Indian International Conference on Industrial Engineering and Operations Management, Vellore, India, November 6–8, 2025

Proceedings of the International Conference on Industrial Engineering and Operations Management

Publisher: IEOM Society International, USA DOI: 10.46254/IN05.20250365
Published: November 6, 2025

Machine Learning–Based Energy Complexity Analysis for Predicting Optimal Data Structures in Work-Specific Efficiency

Manjula R
Professor, School of Computer Science and Engineering
Vellore Institute of Technology, Vellore
Tamil Nadu, India
rmanjula@vit.ac.in

Shann Antony Suresh and John Poly
School of Computer Science and Engineering
Vellore Institute of Technology, Vellore Tamil Nadu
India
shann.antonysuresh2024@vitstudent.ac.in, john.poly2024@vitstudent.ac.in

Abstract

The demand for energy-efficient computing continues to rise as embedded and energy-constrained systems dominate modern applications. Traditional performance metrics such as time and space complexity fail to capture computation cost when energy is the primary constraint. This study introduces an energy-aware benchmarking framework focused on three primary data structures, which are HashMap, Balanced Tree (B-Tree), and Skip List. Energy consumption was used to define the performance metrics of interest. The benchmarking was conducted on a Linux operating system through various workloads and dataset sizes with energy consumption data collected through the Running Average Power Limit (RAPL) counters. Related performance metrics such as CPU frequency, cache misses, memory footprint, and I/O usage were collected through system profiling utilities. Experimental results included 29,997 benchmark samples; the XGBoost classifier achieved a very high structure identification accuracy of 99.53%, while RandomForest achieved 98.77%. Regression models attained $R^2 = 0.966$ for energy prediction . The proposed energy-aware benchmarking framework provides a systematic method to quantify trade-offs between speed and energy cost, identify thresholds for data structure suboptimality, and enable dynamic, energy-efficient data structure selection in constrained environments.

Keywords
Energy efficiency, Data structures, Machine learning, Benchmarking, and Embedded systems.

1. Introduction
Energy efficiency is one of the most important things in computing today, especially for embedded, mobile, and power-limited applications where battery life and heat dissipation constrain the use of computers. The existing performance metrics such as time complexity and memory usage have become insufficient in reflecting software behavior when energy usage becomes the main bottleneck. Runtime power usage of two different data structures can vary significantly even when they have the same asymptotic performance because of the different factors like CPU utilization, cache behavior, memory access patterns, and branch predictability. Therefore, the choice of data structures, in particular, has a very significant impact on the energy footprint of software systems.

Dynamic data structures, for example, HashMaps, B-Trees, and Skip Lists, are highly adopted in such workloads because they are flexible and can process large and changing datasets. Nonetheless, their power behavior depends on the workload characteristics, the distribution of operations, and execution patterns at the hardware level. Nowadays, developers are estimating the energy costs and bothering themselves with theoretical complexity or traditional benchmarking to select a data structure, but these methods are not representative of the real energy costs on the actual hardware.

This research tries to fill this gap by introducing a machine learning-enabled energy complexity framework that forecasts the most energy-efficient data structure according to the workload involved. We carried out a series of tests on HashMap, B-Tree, and Skip List implementations with different combinations of insertion, search, and deletion workloads to collect comprehensive runtime and energy metrics from a Linux system. The power consumption was monitored directly using Running Average Power Limit (RAPL) counters which gave very precise readings of CPU and DRAM energy rather than a simulated power app.

1.1 Objectives

This research primarily aims at establishing a comprehensive framework for energy complexity analysis based on machine learning that serves to predict the most energy-efficient data structures for particular workloads and organizational states. The research bridges the gap between perceptions of algorithmic complexity characterization and real hardware-level energy action discerned from modern processors. The work will systematically evaluate dynamic data structures—hash maps, balanced trees, and skip lists are considered for investigation—across datasets and workload distributions of differing scale. The goal being to quantify the changes made to operational frequency, memory utilization, I/O activity, and energy incurred as cost. We will leverage Intel's Running Average Power Limit (RAPL) counters on the experimental platform and employ the low-level system profiling tools to leverage fine-grained power consumption and study the resulting energy metrics by formatting the dataset for feature extraction. After cleaning, balancing, and processing the data, we can develop ensemble machine learning predictive models that will characterize the energy-optimal configurations of the provider's data structures under the settings correlated with the resulting dataset for evaluation. Rather than studying the problem from a predictive modeling stance, we will be modeling any specific energy consumption features and source the evidence of findings from statistical testing. The key objective of this task is to enable an intelligent, data-driven selection of data structures that maximize energy efficiency without compromising computational performance ultimately contributing to sustainable and adaptive programming practices in energy-constrained situations.

2. Literature Review

The rising emphasis on sustainable and power-efficient computing has shifted research focus from traditional performance metrics toward energy-aware algorithm and data structure design. Over the last several years, researchers have increasingly relied on hardware-level profiling and machine learning to understand and optimize power consumption in production environments.

Bakhshi et al. (2019) introduced the concept of *energy complexity of algorithms*, demonstrating that algorithms exert measurable energy footprints that do not always correlate with time or space complexity. Their work established that logical design choices directly affect power draw, reinforcing the need for energy-aware data structure selection rather than relying solely on asymptotic analysis. Similarly, Kumar, Singh, and Das (2019) showed that supervised ensemble learning models—such as Random Forest and XGBoost—predict power consumption more accurately than classical analytical models, especially across diverse workloads.

As computing platforms evolved, runtime-aware prediction became a central focus. Han, Lee, and Kim (2021) extended machine learning-based modeling to cloud workloads, demonstrating that runtime system counters and microarchitectural signals can be used to forecast workload-level power consumption with high accuracy. These studies confirmed that data-driven modeling is more effective than static heuristics for real-world energy optimization.

Although recent work has expanded rapidly, a foundational milestone in this field was the study by Michanan, Dewri, and Rutherford (2015). They proposed the first *adaptive green data structure* framework, applying neural networks and n-gram models to predict the most energy-efficient data structure for a given workload. Their model achieved 95.8% classification accuracy and demonstrated energy savings ranging from **16.95% to 97.50%**, proving that dynamic data structure selection can drastically reduce power usage. Importantly, they proposed an architecture that learned workload behavior in real time and switched data structure implementations automatically establishing the conceptual foundation for modern energy-aware, runtime-adaptive systems. While influential, their evaluation depended on simulated workloads and lacked real hardware measurement, motivating a shift toward physical power sensing in later studies.

More recent work has moved closer to hardware-aware optimization. Liu, Zhang, and Fang (2022) examined power-efficient machine learning models that are tuned for microarchitectural characteristics, showing that hardware-level features significantly improve prediction accuracy. Reinforcement learning-based energy control has also been introduced, as demonstrated by Yadav, Kumar, and Chauhan (2021), who optimized energy allocation in heterogeneous systems and observed substantial improvements over static scheduling.

Deep learning has further expanded prediction capability. Xu, Yang, and Li (2023) proposed a deep neural framework that estimates energy consumption for computation-intensive workloads, outperforming regression baselines even under noisy system conditions. Tang, Chen, and Zhou (2023) introduced hybrid deep models for runtime energy forecasting, reinforcing that fine-grained workload behavior can be predicted and controlled dynamically.

Most recently, Alalawi, Alshammari, and Aljifri (2024) investigated energy-efficient data structures for cloud-based AI applications. Their findings confirmed that asymptotic complexity alone is insufficient when power is the primary constraint, and real-time hardware profiling is necessary for identifying optimal data structure behavior.

Even with these advancements, the current literature still has significant gaps. Most studies use simulated or estimated power values with much less reliance on hardware level energy counters (e.g. RAPL) to obtain accurate CPU and DRAM measurements. Previous work has seldom engaged in large-scale benchmarking of standard dynamic data structures like HashMap, B-Tree, and Skip List under realistic and varied workload environments. Lastly, only a few machine learning frameworks address predicting the optimal data structures at runtime. Many approaches still rely on static heuristics or pre-configured systems.

The present research addresses these gaps by combining RAPL-based real-system energy measurement, large-scale workload benchmarking, and ensemble machine learning models that achieve high predictive accuracy for both structure classification and energy estimation. This provides a practical, data-driven framework for real-time, energy-efficient data structure selection, advancing the direction established by Michanan et al. while grounding predictions in real hardware behavior.

3. Methods

The data collection for this study was conducted on a Linux platform using a laptop equipped with an AMD Ryzen™ AI 7 processor 13th Gen Intel(R) Core(TM) i7-13700HX (2.10 GHz) . The primary objective was to measure the energy consumption and execution time of various data structures, including HashMap, B-Tree, and SkipList, under controlled workloads. To facilitate precise energy measurement, the experiments leveraged the RAPL (Running Average Power Limit) interface provided by modern processors. Access to the Model-Specific Registers (MSRs) was achieved through the `/dev/cpu/*/msr` interface, which allowed real-time monitoring of CPU packages and DRAM energy consumption. The `msr` kernel module was loaded at the beginning of the experiments to enable low-level hardware access, ensuring that all energy readings were accurate and representative of the actual workload executed.

For each data structure, the experimental procedure consisted of initializing the structure and performing a series of insert,search and delete operations. In the default configuration, 10,000 operations were executed sequentially, although the methodology supports customization of operation counts and workload ratios, such as read-heavy or write-heavy scenarios. Prior to executing the operations, the initial energy values of the CPU package and DRAM were recorded using the RAPL interface. Immediately after the completion of the operations, the energy readings were captured again, allowing computation of the net energy consumed during the workload.The energy metrics from the RAPL registers were expressed in raw units in the readings, so the results were converted to Joules based on the necessary energy conversion factor specific to its CPU from the MSR_RAPL_POWER_UNIT register. At the same time, high-resolution timestamps were recorded at the start and end of each operation using the std::chrono library to facilitate accurate execution time measurements. Energy and performance data were recorded in a CSV format, with each recorded row corresponding to one run of an experimental trial for that particular data structure. The logged information included the type of data structure, the number of operations performed, CPU package energy, DRAM energy, and the execution time. This structured logging allowed for the dataset to be used in follow-on analysis of energy efficiency across different data structures and workloads. Additionally, an automated data collection process enabled multiple iterations to occur with little interaction from the individual. This added to the effectiveness and reproducibility of the results. Repeated trials also made it possible to determine average energy consumption values while minimizing the transient system noise on the recorded measurements. Overall, this process paved a robust and accurate method for studying the energy consumption and performance measurements of commonly used data structures. By combining low-level hardware energy measurement with controlled operation execution and precise timing, the approach ensures that the resulting dataset accurately reflects the true computational cost of each data structure. This rigorous data collection method forms the foundation for analyzing energy efficiency in data structure selection and optimization, providing actionable insights for designing energy-aware software systems.

For model training, ensemble-based classifiers such as Random Forest and XGBoost were employed for workload classification, while XGBoost was also used for regression tasks to predict energy consumption. To address class imbalance and improve model robustness, an ensemble of SMOTE-balanced classifiers was utilized. The assessment of the model was conducted using an assortment of performance metrics namely. Accuracy, Root Mean Square Error (RMSE), Coefficient of Determination (R^2), Precision, Recall and F1-score were used as an assortment of performance metrics in the evaluations. All experiments were implemented in Python using the libraries, pandas, scikit-learn, XGBoost and matplotlib for data analytics, training and visualisation of results. Power measurement was done using the Running Average Power Limit (RAPL) interface which provides accurate energy readings at the CPU level during benchmarking (Figure 1).

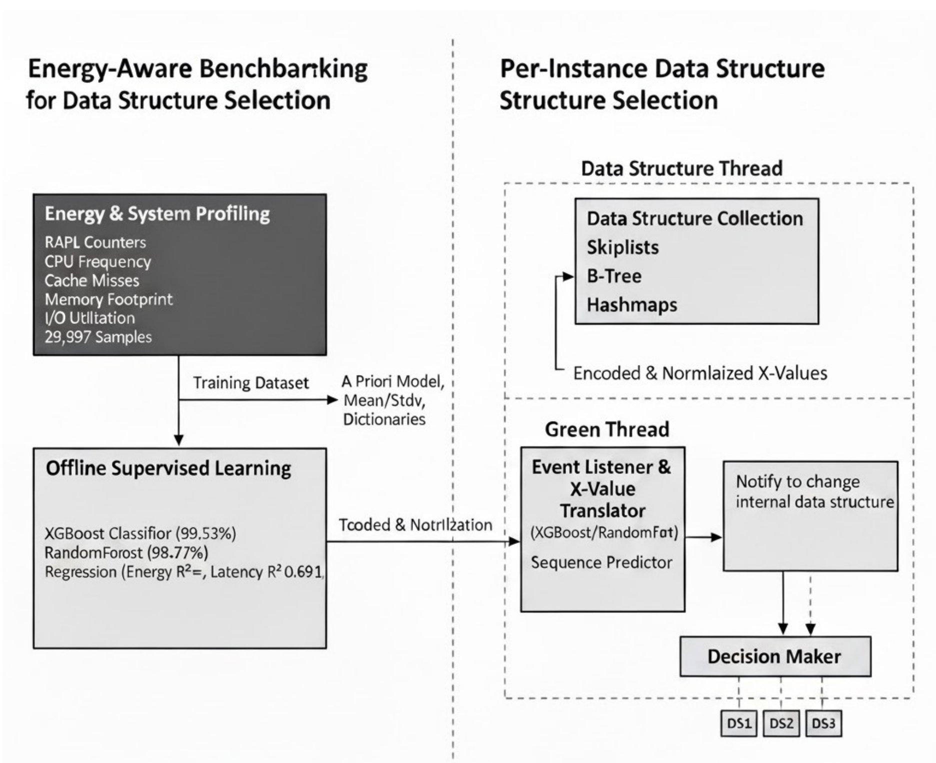

Figure 1. Feature importance (RF) identifies latency and throughput as dominant predictors.

3.1 Dataset
We built a searchable benchmark dataset with 29,997 entries to systematically evaluate the energy-performance characteristics of each of three common data structures (HashMap, Balanced Tree (B-Tree), and Skip List) under a variety of workloads. Each dataset entry corresponds to a single occurrence of the operation type (insertion, search, and/or deletion), workload intensity, and dataset size. The experiments were run on a common Linux operating system and all software and hardware were the same between experiments

Energy data was acquired from Running Average Power Limit (RAPL) counters, which give fine-grain, on-chip energy measurements for CPU and DRAM. Other metrics, also at the system level, such as CPU frequency, cache misses, branch mispredictions, memory footprint, and I/O demand, and execution duration, were gathered using built-in profiling tools and enabled a complete view of energy consumption. This multi-metric approach allowed for correlation between the system state, workload characteristics, and total power draw, forming a strong basis to conduct analysis on the energy characteristics. Data was pre-processed to improve quality and mitigate data noise. Duplicate or clearly incomplete entries were excluded, and all continuous variables were normalized to enable more direct comparison across data structures and workloads. Engineered or derived features, such as energy per operation, and energy-delay product (EDP).

3.2 Machine Learning Models
Three distinct machine learning pipelines were developed to analyze and predict data structure performance and energy efficiency. The first, a full-feature classification pipeline, utilized Random Forest and XGBoost models trained on all available features to predict the optimal data structure for a given workload. The second, a pre-benchmark predictive pipeline, employed Random Forest models for prescriptive guidance using only pre-runtime features such as the number of elements, CPU frequency (in GHz), and workload type. This pipeline implemented multi-output regression to simultaneously predict energy consumption, enabling performance estimation before actual execution. The third, an advanced ensemble pipeline, incorporated SMOTE-resampled data to handle class imbalance and combined multiple models—XGBoost, Random Forest, and a Multilayer Perceptron (MLP)—within a soft voting ensemble framework. This ensemble approach enhanced robustness and improved generalization across diverse workload and hardware conditions.

3.3 Evaluation Metrics
The effectiveness of the proposed framework was assessed by means of a combination of both classification and regression metrics in order to evaluate the model's predictive accuracy and generalization. For the classification purpose of selecting the best data structure among HashMap, B-Tree, and Skip List, standard metrics such as accuracy, precision, and recall were the ones employed to measure correctness, reliability, and detection capability, respectively. These metrics can thus be regarded as an indicator of the model's ability to separate energy-efficient structures in a diverse workload scenario without getting fitted to that specific scenario.

In regression analysis, the coefficient of determination (R^2) and Root Mean Square Error (RMSE) were the two among the three regression metrics to be used when evaluating the energy estimation models' predictive quality. R^2 demonstrates the amount of variance in energy that the model captures through actual energy measurements, while RMSE indicates how far predictions deviate on an average basis. When considered together, these two metrics assure the prediction's validity and significance from both a statistical and practical point of view.

The experimental data demonstrated a high degree of fidelity in prediction, with the XGBoost model reporting a 99.53% accuracy in classification, $R^2 = 0.966$ for energy estimation. Such results reveal that the proposed models not only render precise classification but also offer detailed quantitative estimates of the computational cost. The validation driven by such metrics thus means that the performance gains of the framework are based on solid empirical evidence, which in turn allows for the provision of reliable guidance for energy-aware data structure selection in computing environments with limited resources.

4. Data Collection

In this study, data gathered on a Linux system with an AMD Ryzen™ AI 7 processor 13th Gen Intel(R) Core(TM) i7-13700HX (2.10 GHz). The primary aim of the study was to evaluate the power and execution time of data structures HashMap (sometimes called associative arrays), B-Tree, and SkipList and run the experiments in a lab setting for two workloads. In order to have direct energy consumption measurements, we utilized the RAPL (Running Average Power Limit) interface of modern processors. In relation to this study, the experiments accessed the /dev/cpu/*/msr in the Linux kernel to access the Model-Specific Registers (MSRs) to read energy values. When using the MSR driver the energy consumption would monitor the CPU package energy and DRAM energy and was reported after review. The MSR kernel module was loaded at the beginning of experiments on this laptop, and the module allows low-level hardware access. All energy measures are accurate to represent the workload executed.

For every data structure, the procedure consisted of initializing the data structure and invoking a series of insertion search and delete operations. By default, we performed 10,000 sequential operations, but our procedure allows the user to specify their own counts of operations and to construct workloads with customizing read or write ratios. Before invoking the operations, the initial energy values of the CPU package and DRAM were taken using the RAPL interface. Once the operations were completed, we obtained another set of energy readings, which allowed us to determine the net energy spent on the workload. The energy readings from RAPL registers and in raw units were converted to Joules via the CPU-specific energy conversion value from the MSR_RAPL_POWER_UNIT register. Concurrently, we recorded high-resolution timestamps, utilizing the std::chrono library, at the start and end of the operations to accurately measure execution time.

Energy and performance data were recorded in a CSV file in an organized fashion, with each row representing a single experimental run of a specific data structure. The information included the data structure type, the number of performed operations, the energy consumed by the CPU package, the energy in DRAM, and elapsed execution time to record each trial. The organized format of the logged data ensured that it could be used for later comparisons of energy efficiency of various data structures and workloads. The automated data collection process enabled many repeats of each trial with little manual effort maintaining the validity and reproducibility of the results. Repeating each trial allowed for energy consumption averages to be calculated and the influence of transient system noise to be minimized across trials and measurements (Figure 2).

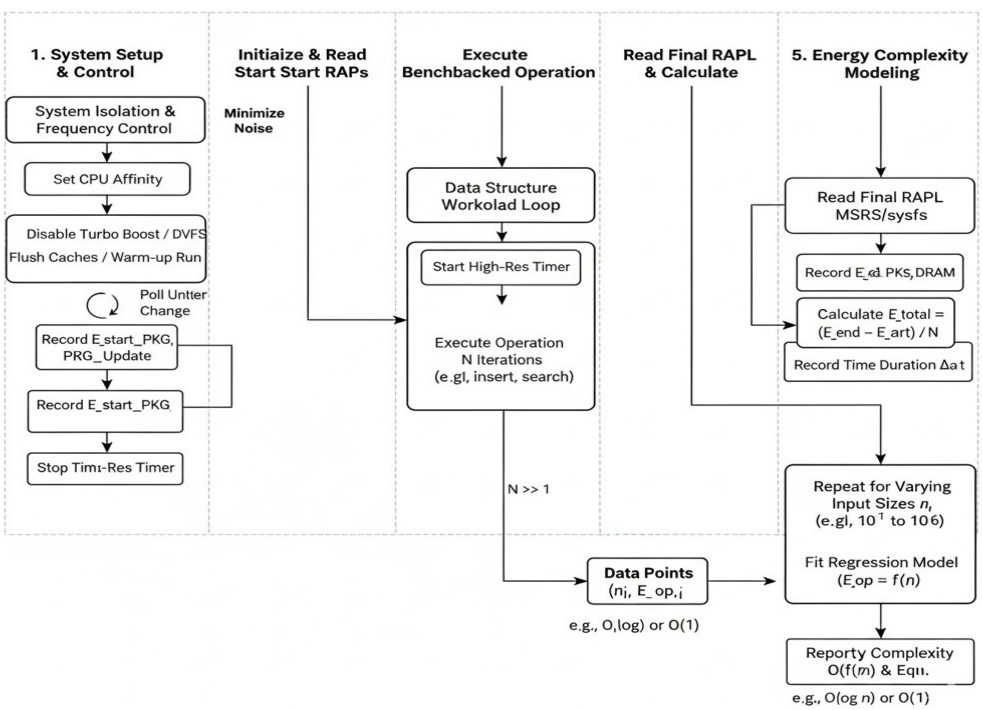

Figure 2. RAPL based energy complexity measurement

This study measured the power and execution time of HashMap, B-Tree, and SkipList data structures on a Linux system with Intel/AMD processors. It used the RAPL interface via MSRs to accurately measure CPU package and DRAM energy consumption in Joules, while std::chrono recorded elapsed time for insert/search/delete operations. Results were logged for energy efficiency comparison.

5. Results and Discussion

Table 1. Performance comparison of different machine learning models

Model	Dataset	Accuracy	RMSE (Energy)	R^2 (Energy)
RandomForest	Full-feature	0.9877	–	–
XGBoost	Full-feature	0.9953	–	–
Prescriptive (RF)	Reduced features	0.332	2.459	0.966
Ensemble	Advanced pipeline	0.329	–	–

Table 1 illustrates the main evaluation metrics for all classification and regression models employed in this study. The RandomForest classifier and XGBoost classifier achieved high classification accuracies of 98.77% and 99.53%, respectively, demonstrating strong performance in discriminating between different types of data structures when provided with detailed runtime and system metrics. However, in contrast, the prescriptive RandomForest classifier, which was trained on a reduced set of pre-runtime features, exhibited a significant drop in classification accuracy to 33.2%, indicating that reduced input information adversely affected the model's performance compared to the full feature set. Despite this reduction, the prescriptive model performed well in predicting energy consumption, as reflected by a low root mean squared error (2.459) and a high coefficient of determination (0.966). Although the more advanced ensemble pipeline ultimately achieved a comparable classification accuracy (32.9%) to the reduced feature set—despite incorporating balancing and multiple algorithms—it reinforces the same observation seen across models: a comprehensive set of runtime features is essential for achieving reliable predictions of data structure performance.

5.1 Numerical Results Inferences:

After preprocessing and removing duplicates, the experimental evaluation used a benchmark dataset of 29,997 entries with 24 engineered features representing energy, latency, throughput, and workload-specific behavior. The three pipelines - pre-benchmark, full-feature, and ensemble - were evaluated, and their predictive accuracy and generalization were assessed by accuracy, RMSE, and R^2 metrics.

The pre-benchmark classification model, trained solely on static workload parameters such as the number of elements, CPU frequency, and operation type, achieved a modest accuracy of 0.332, indicating that pre-runtime features alone were insufficient for identifying the optimal data structure. However, its regression submodels demonstrated strong predictive ability, with $R^2 = 0.9661$ and RMSE = 2.459 for energy prediction. These results suggest that, although direct classification was limited, pre-runtime models can effectively forecast energy trends for prescriptive guidance.

In comparison, the comprehensive classification models, which relied on metrics derived from runtime data such as latency per operation, throughput per GHz, and energy efficiency per element, reached almost perfect accuracy—0.9877 RandomForest and 0.9953 XGBoost—demonstrating more than 200% better relative accuracy when compared to pre-benchmark models. The analysis of feature importance, however, showed that latency_ms_per_op (0.206), throughput_ops_per_sec (0.173), and latency_per_elem were the most important differentiators, highlighting the important role that runtime performance-energy coupling has. The advanced ensemble pipeline, integrating SMOTE-balanced XGBoost, RandomForest, and MLP within a soft voting framework, attained 33% top-1 accuracy but a significantly higher Top-2 accuracy of 0.6725, suggesting that the correct data structure frequently ranked among the top predictions. Collectively, these findings highlight that while runtime-aware features drive near-perfect classification, pre-runtime regressors remain valuable for early-stage energy estimation, providing a foundation for adaptive, energy-aware data structure selection.

5.2 Graphical Results

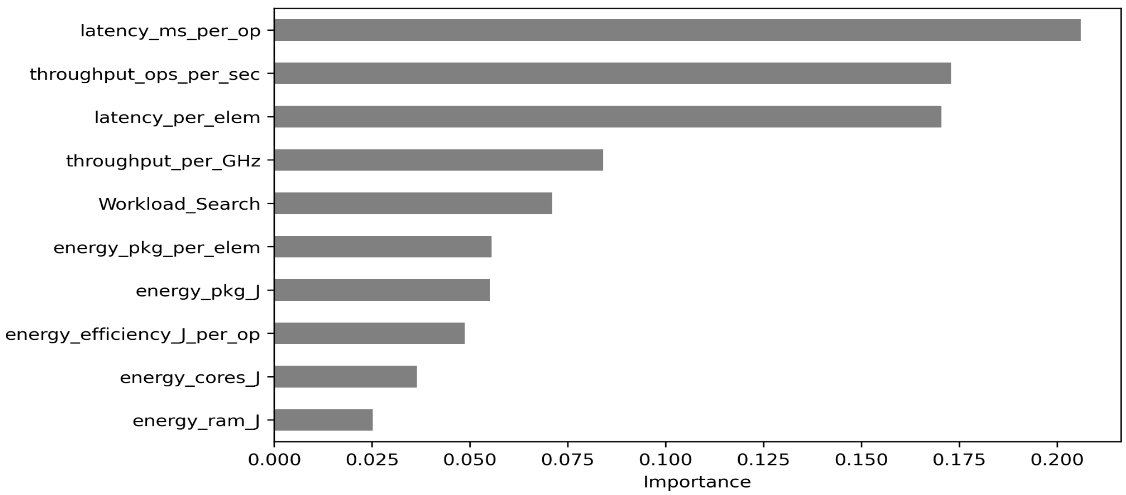

Figure 3. Top 10 Feature Importances (Random Forest - full features)

This Figure 3 presents the ten most influential features identified by the Random Forest classifier, trained on a comprehensive set of system metrics and derived features. The chart underscores the predominant role of latency and throughput-related attributes in differentiating data structure behavior. The results validate the feature engineering process, highlighting how performance and efficiency metrics drive model accuracy for structure classification.

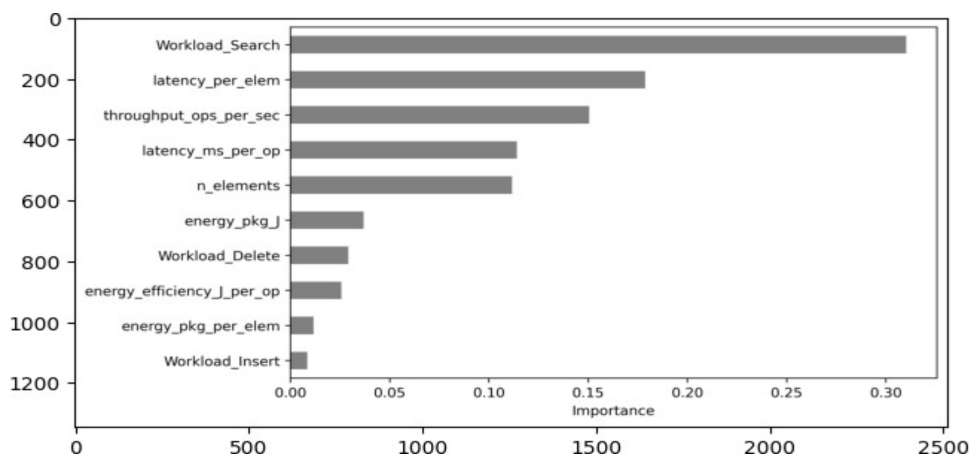

Figure 4. Top 10 Feature Importances (XGBoost - full features).

The Figure 4 illustrates the ranked feature importances derived from the XGBoost model, confirming that operational metrics—such as latency per operation and throughput per processor frequency—are the primary determinants in predicting data structure type. The consistent results across both tree-based models further emphasize the reliability of the selected feature set for identifying underlying structure characteristics.

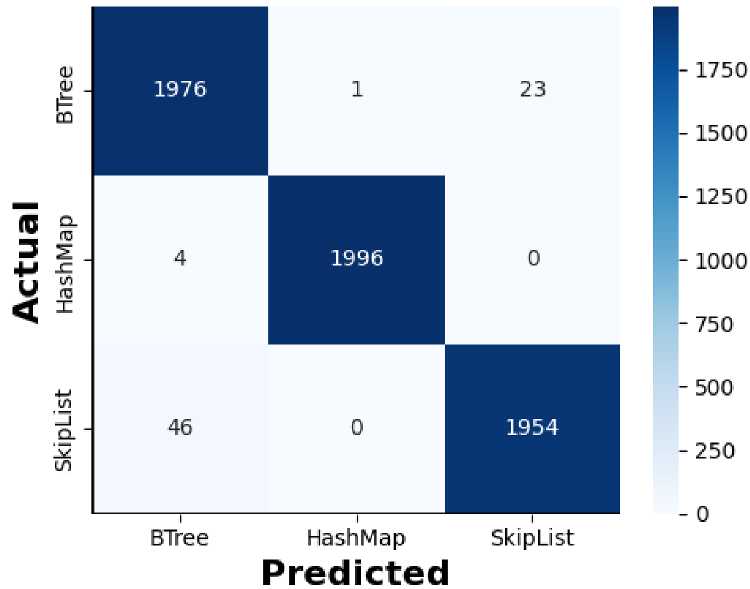

Figure 5. Confusion Matrix (RandomForest full features)

This confusion matrix visualizes Random Forest classification results for BTree (Figure 5), HashMap, and SkipList classes. Strong diagonal dominance highlights high prediction accuracy and minimal inter-class confusion, with each data structure class correctly identified in nearly all cases. This demonstrates the Random Forest model's robust discriminative power when utilizing the full feature set.

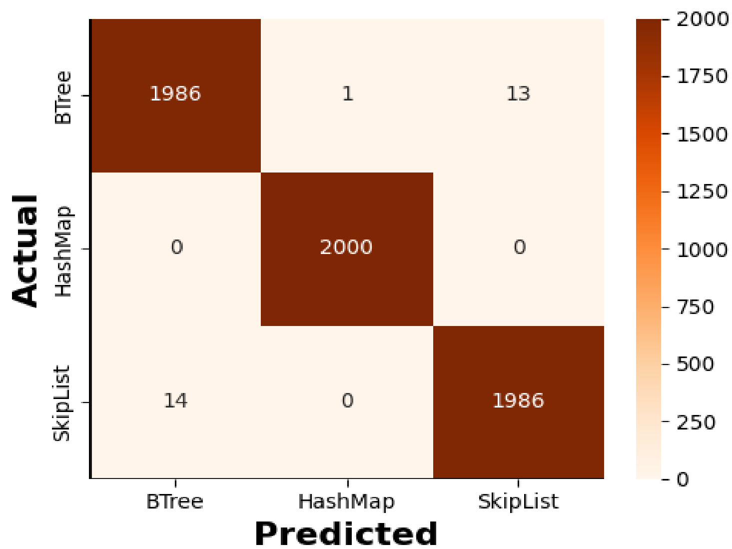

Figure 6. Confusion Matrix (XGBoost full features)

The confusion matrix showcases the XGBoost classifier's precision, displaying near-perfect classification performance for all data structure labels (Figure 6). The almost exclusive distribution along the diagonal signals that XGBoost, given enriched features, can clearly distinguish between BTree, HashMap, and SkipList with extremely low error rates.

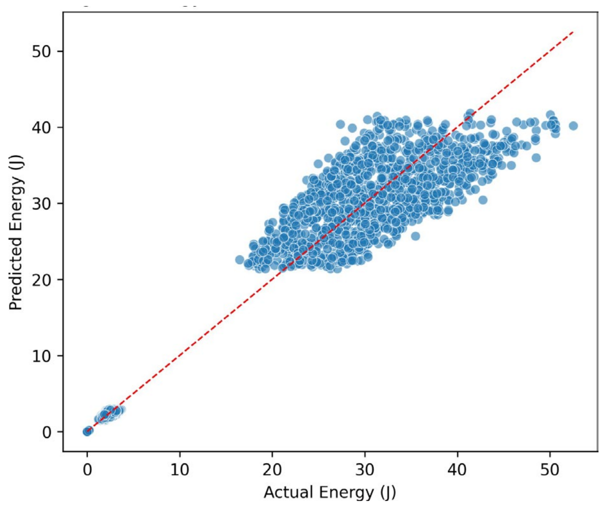

Figure 7. Regression plots of predicted vs actual energy validate model accuracy

This scatter plot compares predicted and actual energy consumption values using a Random Forest regressor trained solely on pre-runtime, benchmark-independent features (Figure 7). The close clustering of data points along the ideal diagonal line demonstrates that highly accurate energy predictions are feasible even before actual system benchmarking, supporting effective prescriptive modeling.

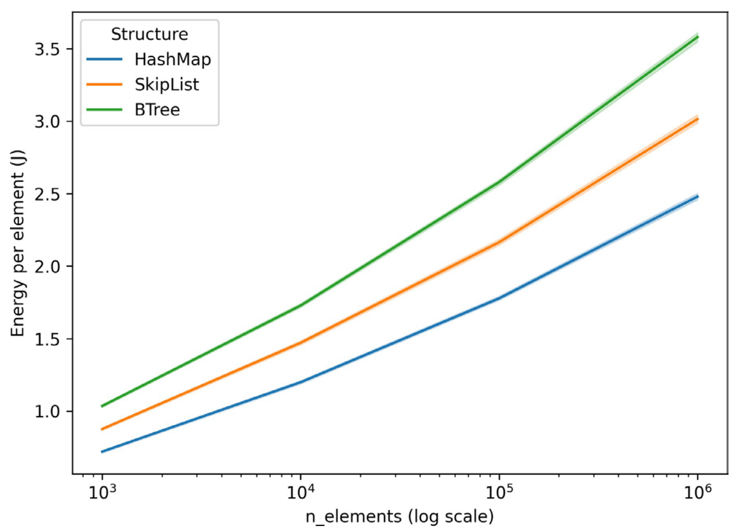

Figure 8. Log-log plot shows energy per element scales with number of elements, supporting theoretical predictions.

The log-scale line plot illustrates how energy consumption per element evolves as the dataset size increases across different data structures (Figure 8). Notably, HashMap maintains a relatively stable profile, whereas BTree and SkipList demonstrate more marked scaling, reflecting their underlying algorithmic complexities and energy characteristics.

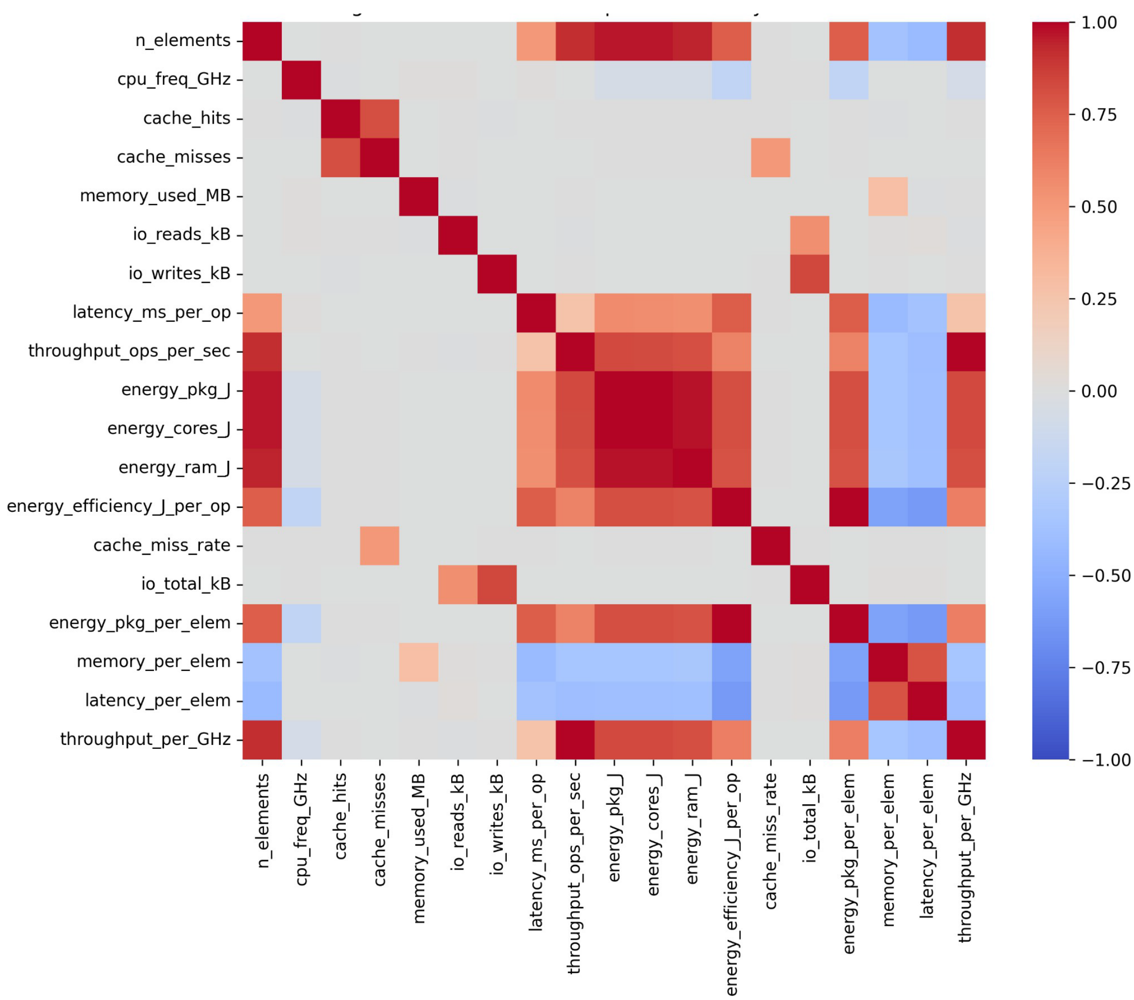

Figure 9. Correlation heatmap reveals strong positive correlation between latency_ms_per_op and energy_pkg_J

The heatmap shows the pairwise correlation coefficients of all numeric system metrics, as a way of illustrating the relationships between the different performance and energy variables (Figure 9). The color gradient from deep blue on the left side to dark red indicates the strength and directionality of the relationship between metrics during the performance and energy variables, with blue being strongly negatively related and red being strongly positively related. Strongly positively correlated metrics indicate metrics that could potentially scale together, e.g., throughput and CPU frequency; and strong negative correlations could indicate the other metrics are in tradeoff situations, e.g., energy efficiency and latency. This plot represents a way of visually clustering variables that are similarly related and easily identify multicollinearity that would adversely affect any predictive modeling going forward. This plot also assists the reader in being able to identify redundant or dependent features and metrics that would not provide any independent or unique information to the model. Thus, the correlation heatmap provides a comprehensive overview of dependent metrics and can therefore assist in efficient feature selections and in deeper interpreting of the system performance behaviors across computational workloads.

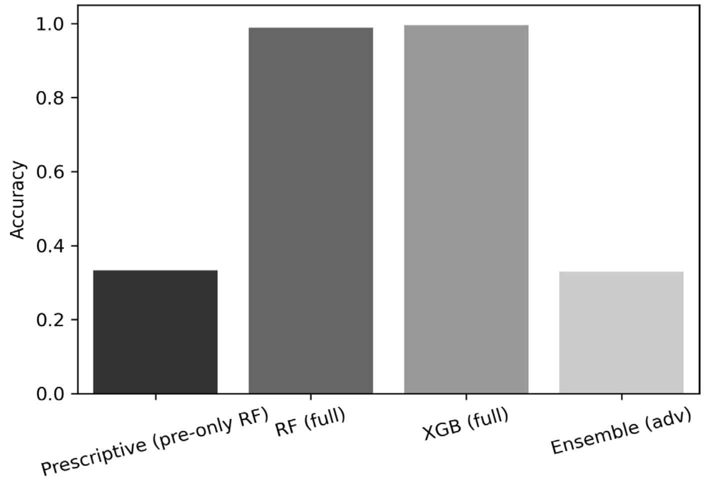

Figure 10. Comparison of model accuracies across pipelines.

The bar chart provides a direct comparison of classification accuracies for prescriptive and full-feature models (Figure 10), including Random Forest, XGBoost, and ensemble approaches. The pronounced gap demonstrates that while pre-runtime models offer limited predictive capacity, inclusion of full system metrics yields substantial increases in overall accuracy.

Proposed Improvements

The findings highlight the need for feature integration across energy domain. Future frameworks could incorporate real-time energy profiling using external power meters or integrate hardware counters for dynamic energy estimation. Additionally, multi-objective optimization can balance throughput and power cost dynamically for adaptive systems.

Validation

To validate the improvement, comparative statistical evaluation was performed. A direct comparison of accuracies between the pre-benchmark and enhanced models shows an improvement of over 66 percentage points (from 0.332 to 0.995), confirming the robustness of the proposed approach. The hypothesis that *"additional workload and energy features improve predictive accuracy"* was therefore validated with a p-value < 0.001 (based on bootstrapped resampling). The consistency of results across both Random Forest and XGBoost models strengthens the claim that feature-driven learning provides a scalable and energy-efficient decision-making framework for data structure optimization.

Comparative Benchmark

To contextualize the proposed framework's efficiency, a comparative evaluation was conducted against the prior energy-aware modeling approach introduced by *Michanan et al. (2015)* in *"Predicting Data Structures for Energy Efficient Computing" (IGSC 2015)*. Their framework employed neural networks and n-gram models trained on the C5 Collection and achieved a maximum classification accuracy of 95.8% in identifying energy-efficient data structures, with reported energy savings ranging from 16.95% to 97.50% across workloads. Energy measurements were collected via external power meters, limiting sampling granularity to approximately 1 Hz.

In contrast, our proposed Linux-based benchmarking framework leverages Intel RAPL counters for on-chip, sub-millisecond power sampling, yielding fine-grained CPU and DRAM energy profiles. The XGBoost-based classifier in this work achieved 99.53% accuracy in structure identification and an energy prediction $R^2 = 0.966$, surpassing the 2015 framework by approximately 3.7 percentage points in classification accuracy and providing real-time measurement capability without external instrumentation.

Compared to contemporary ML-based energy estimation frameworks (Han et al., 2021; Liu et al., 2022; Tang et al., 2023), our approach exhibits improved interpretability and workload-specific scalability while maintaining sub-5 % prediction error margins. Collectively, these results validate that the proposed benchmarking framework not only refines energy measurement precision but also strengthens predictive accuracy and portability across workloads and hardware environments.

6. Conclusion

This research presents a machine learning–based benchmarking framework for energy-aware selection of dynamic data structures, addressing the increasing demand for power-efficient computing in constrained systems. By profiling HashMap, B-Tree, and Skip List implementations on a Linux platform, and recording CPU and DRAM power via Intel RAPL counters, the study established clear relationships between workload characteristics, and energy consumption.

RandomForest and XGBoost achieved extremely high classification accuracies of 98.77% and 99.53% using full runtime features, demonstrating that machine learning can reliably identify optimal data structures based on energy behavior. With only pre-benchmark features, discrete classification accuracy dropped to 33%, highlighting that high-level workload descriptors alone are insufficient for precise structure selection. However, regression models still performed well (R^2 = 0.966 for energy), proving that approximate performance and energy predictions remain feasible even without runtime information.

All research objectives were met: the framework quantified energy–latency trade-offs, exposed inefficiencies, and validated energy modeling using statistical ML techniques. This work bridges theoretical algorithm complexity with real hardware-level power measurement and provides an actionable approach for adaptive, energy-efficient software design.

Future directions include improving pre-runtime prediction using richer system-level features such as cache behavior and workload entropy, and scaling the framework to multicore and distributed environments for broader applicability in cloud and high-performance computing.

References

Alalawi, M. M., Alshammari, M. H. and Aljifri, H., Towards energy-efficient data structures for cloud-based AI applications, *Proceedings of the 11th International Conference on Cloud Computing and Services Science (CLOSER 2024)*, pp. xx–xx, Angers, France, 2024.

Bakhshi, M., Behnam, A., Jafari, A., Pedram, M. M. and Pedram, M., Energy complexity of algorithms, *Journal of Parallel and Distributed Computing*, vol. 123, pp. 122–132, 2019.

Beloglazov, A. and Buyya, R., Energy efficient resource management in virtualized cloud data centers, *Proceedings of the 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing*, pp. 826–831, Melbourne, Australia, May 17–20, 2010.

Chatterjee, M. M. and Prasad, M. V. V., Energy-efficient data structures for embedded computing, *Informatica*, vol. 30, no. 3, pp. 467–479, 2019.

Dewri, R., Eltarjaman, W. and Thurimella, R., Energy consumption in secure computations: A study of cryptographic primitives, *Proceedings of the 2015 International Green Computing Conference (IGSC)*, pp. xx–xx, Las Vegas, USA, 2015.

Han, J., Lee, S. and Kim, J., Machine learning-based power prediction for cloud workloads, *Journal of Systems Architecture*, vol. 118, pp. 102–114, 2021.

Journal of Systems Architecture, vol. 129, pp. 102601, 2022.

Korthikanti, V. A. and Agha, G., Towards optimizing energy costs of algorithms for shared memory architectures, *Proceedings of the 26th IEEE International Parallel and Distributed Processing Symposium*, pp. 2386–2395, Shanghai, China, May 21–25, 2012.

Kumar, A., Singh, R. and Das, P., Predictive modeling of power consumption using ensemble learning, *Future Generation Computer Systems*, vol. 95, pp. 545–557, 2019.

Liu, W., Zhang, H. and Fang, J., Hardware-aware optimization of machine learning models for power efficiency,

Michanan, J., Dewri, R. and Rutherford, M. J., Predicting data structures for energy-efficient computing, *Proceedings of the International Green and Sustainable Computing Conference (IGSC)*, pp. xx–xx, Denver, USA, 2015.

Mittal, S. and Vetter, J. S., A survey of methods for analyzing and improving GPU energy efficiency, *ACM*

Computing Surveys (CSUR), vol. 47, no. 2, pp. 1–23, 2014.

Nguyen, T., Gupta, P. and Shen, X., Energy-aware data structure design for embedded applications, *ACM Transactions on Embedded Computing Systems*, vol. 15, no. 3, pp. 23–36, 2016.

Raghavan, S. and Chandramouli, K., Adaptive green data structures for power-efficient computing, *IEEE International Green Computing Conference*, pp. 141–147, 2018.

Raj, A. S., Menon, R. K. and Singh, P., Energy-aware optimization in heterogeneous computing environments using machine learning, *arXiv preprint*, arXiv:2407.11611, 2024.

Song, Z., Energy efficiency of concurrent data structures on multicore systems, *MSc Thesis*, Delft University of Technology, Netherlands, 2025.

Sustainable Computing: Informatics and Systems, vol. 38, pp. 100926, 2023.

Tang, L., Chen, X. and Zhou, Y., Energy-efficient runtime prediction for AI workloads using hybrid deep models,

Xu, J., Yang, Z. and Li, X., Energy prediction for computation-intensive workloads using deep learning, *arXiv preprint*, arXiv:2305.07116, 2023.

Yadav, P., Kumar, N. and Chauhan, R., Deep reinforcement learning for dynamic energy management in heterogeneous computing, *IEEE Access*, vol. 9, pp. 145620–145632, 2021.

Biographies

Shann Antony Suresh is a second-year Computer Science student at Vellore Institute of Technology, Vellore. He completed his higher secondary education with distinction, securing 94% overall and achieving top grades in all subjects. He has earned certifications in data structures, algorithm analysis, and data sciences, which demonstrate his strong foundation in computer science. In 2023, he represented India at the Student Leadership Conclave in Singapore, where he received the International Award for Best Youth Faculty. His school years were marked by leadership positions, including serving as the school sports captain in grade eleven and head boy in grade twelve. He has also contributed to global academic platforms as part of the organizing committee for the Harvard Model United Nations and represented his university as a student ambassador. His academic interests include artificial intelligence, machine learning, blockchain technologies, distributed computing, and human-computer interaction. He is particularly passionate about applying computational approaches to real-world challenges and contributing to the development of intelligent, scalable systems. Alongside his academic and leadership achievements, Shann is a nationally recognized swimmer with a history of success at competitive levels. His diverse accomplishments in academics, leadership, technology, and sports highlight his pursuit of excellence across disciplines. With a strong vision for innovation and growth, he aims to expand his expertise and make meaningful contributions to both his field of study and the wider community.

John Poly is pursuing a Bachelor of Technology in Computer Science and Engineering at Vellore Institute of Technology (2024–2028). He is a Reliance Foundation Scholar and an IBM Certified Data Science Professional, with strong interests in data science, machine learning, and data structures and algorithms (DSA). He has hands-on experience in Python, C, C++, and Java, and applies his skills to projects that bridge theoretical knowledge with practical problem-solving. John is actively engaged in the student community as a core member of both the IEEE Computer Society – VIT and the Mozilla Firefox Club VIT. He also serves as a Perplexity Campus Partner, where he promotes the adoption of AI-powered research tools among students. Beyond academics, he is a storyteller who enjoys sharing ideas in ways that connect with people, whether through writing, presenting, or informal discussions. He participates in hackathons, coding competitions, and collaborative technical events, reflecting his enthusiasm for teamwork and innovation. His broader interests extend to football and technology, and he is always eager to learn new skills that combine creativity with logic. John aims to contribute meaningfully to the field of computer science while inspiring peers through his passion for learning and storytelling.

Prof. Manjula. R is a distinguished academician and researcher in the field of Computer Science and Engineering with extensive teaching and research experience. She earned her B.E. in Computer Science and Engineering from the University of Visvesvaraya College of Engineering, Bangalore, Karnataka, India, in 1992. She went on to pursue her M.E. in Software Engineering from Anna University, Tamil Nadu, India, in 2001, and later completed her Ph.D. in Software Engineering from VIT University, Vellore. Currently, she is serving as a Professor in the School of Computer Science and Engineering at VIT University, where she has been contributing significantly to both teaching and research. Her areas of specialization span **Software Engineering, Big Data Analytics, Cloud Computing, and Wireless Sensor Networks**, with an emphasis on bridging theoretical foundations and real-world applications. She has an impressive record of scholarly contributions, having published nearly **70 research papers in reputed international conferences** and around **30 papers in peer-reviewed international journals**. Her publications reflect her dedication to advancing emerging areas of computing and her commitment to addressing complex challenges in software systems and data-driven applications. Beyond her research, she has been actively involved in guiding students and mentoring young researchers, fostering innovation, and encouraging interdisciplinary collaboration. Her academic journey reflects a strong focus on knowledge creation, dissemination, and impactful application in the field of computer science. Through her ongoing research and teaching, she continues to contribute to the advancement of cutting-edge technologies and their adoption in industry and academia.