Comparison of Cloud-Based Free-Tier Virtual Environments: A Performance Analysis on Machine Learning Training

Marlon Juhn M. Timogan[1], Ariel Roy L. Reyes[2],
Saint Michael College of Caraga, Nasipit, Agusan del Norte, Philippines [1]
University of Southeastern Philippines, Davao City, Philippines[1,2]
mjmtimogan01202401714@usep.edu.ph

Abstract – The increasing demand for machine learning (ML) across various domains has driven the need for accessible training environments, particularly for researchers and students lacking high-end hardware. Cloud-based platforms such as Google Colaboratory and Kaggle Notebooks offer free-tier access to computational resources, making them popular options for model development. However, limited comparative research exists on their real-world performance when training ML models. This study introduces a systematic, experiment-driven benchmarking framework that directly compares these two platforms under identical conditions. The key idea of this approach is to evaluate platform efficiency by training both deep learning (Convolutional Neural Networks) and traditional machine learning (Decision Trees) models on a standardized dataset (CIFAR-10), while capturing quantitative metrics such as training time, memory usage, and CPU utilization. Unlike prior studies, which focus on individual platform capabilities, this work provides a side-by-side, reproducible comparison that reveals how platform design impacts performance for different model types. Results show that Kaggle Notebooks outperform Google Colaboratory, achieving 62% faster training for CNNs and 38% faster for Decision Trees, with lower memory and CPU usage. The findings contribute new insights for students, researchers, and practitioners when choosing cloud-based free-tier platforms for machine learning development.

Keywords – Cloud computing, Machine Learning Training, Google Colaboratory, Kaggle Notebooks, Performance Benchmarking.

1 Introduction

The widespread adoption of machine learning (ML) in domains such as healthcare, finance, education, and scientific research has intensified the need for accessible and efficient tools for model development [1]. While modern ML frameworks, such as TensorFlow and PyTorch, have lowered the barrier to entry for training ML models, intense neural networks, and substantial computational resources are still required, including high-performance CPUs and GPUs [2]. For students and independent researchers, cloud-based free-tier platforms provide a practical alternative for experimenting, testing, and developing machine learning models, particularly for resource-intensive machine learning tasks.

Two of the most prominent free-tier platforms today are Google Colaboratory (Colab) and Kaggle Notebooks. These platforms provide browser-based coding environments with built-in support for Jupyter Notebooks, Python libraries, and data science workflows [24]-[27]. Google Colab is integrated with Google Drive, allowing users to execute notebooks with free access to GPUs such as the NVIDIA Tesla T4 and P100, subject to session time and usage limits [3]. Kaggle Notebooks, hosted on the Kaggle platform, prioritize stability and reproducibility, offering seamless access to hosted datasets and competition kernels. While GPU access is also available, it is more limited in duration and availability compared to Colab [4].

Despite the widespread use of Colab and Kaggle Notebooks, a notable gap exists in the literature regarding their direct performance comparison. No studies have rigorously evaluated free-tier ML environments side by side under identical conditions. Most prior research has either examined individual platforms in isolation or focused on paid cloud services, leaving the performance characteristics of free-tier tools underexplored. For instance, Carneiro et al. [1] benchmarked Colab's GPU performance for deep learning tasks but did not compare it against other platforms, highlighting Colab's limitations (such as session timeouts and lack of persistent storage) only in a single-platform context.

Other studies have evaluated cloud providers such as AWS, Google Cloud, and Azure in terms of training workloads. Still, those comparisons typically omit community free services like Colab and Kaggle and often overlook reproducibility across different model types [5], [6]. This lack of comprehensive, side-by-side evaluation means that practitioners and researchers currently must rely on anecdotal experience or incomplete information when choosing a free platform, which can lead to inefficient experimentation and unexpected failures.

To address this open problem, this study conducts a systematic, experiment-driven benchmarking framework for cloud-based free-tier ML environments. The researchers conduct a rigorous head-to-head evaluation of Google Colab and Kaggle Notebooks under identical experimental conditions, thereby strengthening the reproducibility and fairness of the results. Both platforms are compared based on metrics such as training time, memory usage, and CPU utilization.

Generally, this study aims to provide a comparative analysis of Google Colab and Kaggle Notebooks for ML training tasks and recommend a suitable free-tier environment based on resource demands and model complexity.

2 Related Literature

This section synthesizes existing studies on machine learning training environments, cloud-based infrastructures, and performance benchmarking, highlighting limitations addressed by the present study.

2.1 Machine Learning and Training Environments

Machine learning (ML), a subset of artificial intelligence, enables systems to learn from data and is widely applied in image classification, speech recognition, and medical diagnostics [1], [2]. Training ML models, particularly deep learning architectures such as Convolutional Neural Networks (CNNs), requires substantial computational resources, underscoring the importance of efficient and scalable training environments [3].

ML training environments are generally classified as local or cloud-based. Local setups offer control, privacy, and offline accessibility but are limited by fixed hardware capacity [4]. Cloud-based platforms such as Google Colaboratory, AWS, and Kaggle Notebooks provide scalable computing resources with GPU/TPU acceleration and simplified setup [4]–[6]. However, these platforms introduce trade-offs, including internet dependency, session limits, and potential costs for extended usage.

2.2 Serverless and Distributed ML Frameworks

Ali et al. [2] proposed a serverless ML training framework that achieved up to 8 times faster training and 3 times cost efficiency compared to traditional virtual machines. However, limited GPU support remained a constraint. Similarly, Sarroca and Sánchez-Artigas [3] introduced MLLess, reporting up to 15× speedup and 6.3× cost reduction, but with reduced flexibility due to architectural limitations. Pakdel and Herbert [6] demonstrated cost savings using adaptive cloud resource selection, although their evaluation was limited to heterogeneous workloads.

Federated and decentralized learning models further aim to reduce centralized computation and communication overhead. Guerra et al. [7] showed that gossip-based federated learning minimized resource usage, while Teixeira et al. [8] reported up to 83% faster training in decentralized FL settings. However, concerns remain regarding energy consumption, scalability, and real-world deployment. Rajagopal et al. [9] integrated federated learning with blockchain for healthcare applications but identified increased latency and power consumption as challenges.

2.3 Performance Benchmarking of Cloud Platforms

Carneiro et al. [1] benchmarked Google Colaboratory for CNN training and found performance comparable to mid-range GPUs, validating its suitability for educational and lightweight ML tasks, though scalability and session limitations restricted professional use. Lawrence et al. [10] compared TensorFlow performance across local and cloud environments, noting speed advantages for local GPUs and flexibility benefits for cloud platforms. Chahal et al. [5] compared major cloud providers and reported trade-offs between performance and cost. Still, free-tier platforms, such as Kaggle Notebooks, were not included, and advanced configurations, like TPUs, were not evaluated [11]–[13].

2.4 Cloud-Based ML Applications and Workflow Design

Cloud-based ML has been applied across healthcare, manufacturing, transportation, and finance, achieving high accuracy and scalability [14]–[19]. However, many systems rely on proprietary tools, lack interpretability, or face challenges related to security and deployment. Studies on ML architecture and workflow optimization emphasize elasticity, automation, and MLops integration [20]–[23], yet issues related to reproducibility, privacy, and fault tolerance persist.

3 Methods

This section presents the conceptual framework of this study, including the selection of machine learning models, the distinction of the cloud-based free-tier training environments, and the evaluation of performance and computational efficiency.

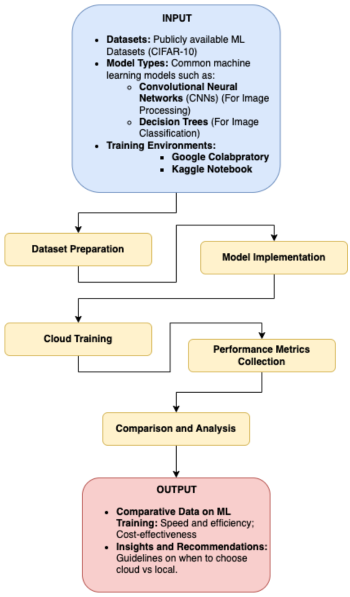

Fig. 1. Conceptual Framework of the Study

Fig. 1 shows the study's conceptual framework, which is structured using the input-process-output (IPO) model. The input stage includes the publicly available machine learning dataset, the CIFAR-10. It also includes selected model types, such as Convolutional Neural Networks (CNNs) and Decision Trees. The two training environments are Google Colaboratory and Kaggle Notebook.

4 Results & Discussion

This section provides an overview of the study results, detailing the methodologies applied, findings obtained, and interpretations for each component of the machine learning training environments evaluated. The study is conducted in phases: Phase 1: Dataset Preparation, Phase 2: Training Setup, Phase 3: Training Execution, Phase 4: Metrics Collection, Phase 5: Analysis and Comparison, and Phase 6: Results Interpretation and Recommendation.

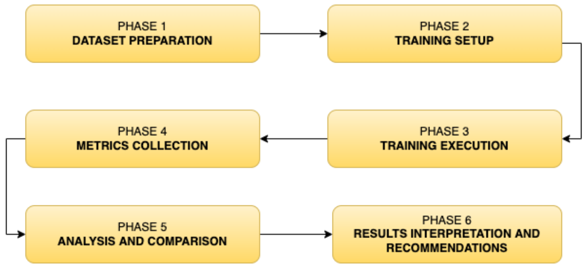

Fig. 2. Phases of the Study

Fig. 2 presents the six phases of the study: dataset preparation, training setup, training execution, metrics collection, analysis and comparison, and results interpretation with recommendations for selecting an appropriate cloud-based free-tier training environment.

4.1 Dataset Preparation

The CIFAR-10 dataset, as shown in Fig. 3, is utilized due to its balanced classes, manageable size, and established reputation as a benchmark for image classification. It consists of 60,000 labeled 32×32 color images across 10 classes and is suitable for evaluating both CNN and Decision Tree models. The dataset is cleaned and preprocessed through normalization, label encoding, and data augmentation to improve model generalization.

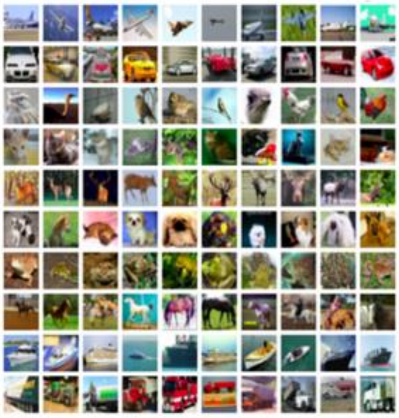

Fig. 3. CIFAR-10 dataset sample images. Retrieved from: https://www.kaggle.com/c/cifar-10/

4.2 Training Setup

The technical groundwork for training a machine learning model is done in this phase. Models will be trained on the cloud using the same configurations with the highest available hardware option. This ensures that performance comparison is accurate and fair. Below is the discussion on the environment configuration, model definition, script finalization, and consistency validation.

A. Environment Configuration

The cloud-based free-tier environments are prepared to ensure consistency in running the machine learning experiments. Google Colaboratory and Kaggle Notebooks are utilized for their accessible GPU acceleration. The cloud environments are also set up with the identical versions of ML libraries and configurations to mirror the local setup as closely as possible, ensuring a fair and accurate performance comparison across platforms.

Table 1. Training Environments – Hardware Specifications

Training Environment	RAM (GB)	CPU (GHz)	Disk (GB)
Google Colaboratory	12.7	2.2	107.7
Kaggle Notebook	30	2.2	57.6

Table 1 summarizes the hardware specifications of Google Colaboratory and Kaggle Notebooks. Google Colaboratory provides 12.7 GB of RAM with larger disk space, but it has session and storage limitations. In contrast, Kaggle Notebooks offers higher RAM at 30 GB with a similar CPU speed. Both platforms were tested under identical conditions to ensure a fair comparison of their performance.

B. Model Definition

The study employs CNNs for complex image classification and Decision Trees for simpler tasks. Model architectures and training components, including loss functions, optimizers, and hyperparameters, are consistently defined across environments to ensure uniform and fair performance evaluation.

C. Script Finalization

Training scripts are written in Python to compile models, execute training, and log key metrics, including accuracy, loss, and training time. Monitoring tools are integrated to track CPU/GPU usage, as well as memory consumption, with file paths configured to ensure compatibility across both local and cloud environments.

D. Validation of Consistency

Preliminary test runs are conducted in both environments to verify the correct loading of data, execution of the model, and consistent behavior. Identical random seeds and initial outputs are checked to ensure reproducibility and comparable processing across platforms.

4.3 Training Execution

In this phase, CNN and Decision Tree models are trained on Google Colaboratory and Kaggle Notebook using the preprocessed CIFAR-10 dataset. Identical configurations, random seeds, and training parameters are applied to ensure reproducibility, while real-time monitoring tools record training time, resource utilization, and execution logs.

4.4 Metrics Collection

This phase systematically collects performance metrics during and after training machine learning models across all computing environments. This phase ensures that data used for comparison is accurate, consistent, and representative of actual resource utilization and computational behavior. Key metrics collected include training time (measured in seconds), memory consumption (expressed as a percentage of total capacity), and CPU/GPU utilization (also expressed as a percentage of total capacity). These metrics are captured using system monitoring tools, such as psutil, as well as environment-specific dashboards.

A. Training Time

This study defines training time as the total duration a model takes to complete all training epochs, measured from the start of the training process to its completion. The training time was recorded using Python's built-in time module, which logged the beginning and end times of each session. Tables 2 and 3 show the training time for the CNN and the Decision Tree Model.

Table 2. Training Time – CNN Model

Training Environment	Training Time (s)
Google Colaboratory	7529.15
Kaggle Notebook	2843.95

Table 3. Training Time – Decision Tree Model

Training Environment	Training Time (s)
Google Colaboratory	238.19
Kaggle Notebook	146.83

B. Memory Usage

The memory behavior of each platform was observed throughout the training process to capture both average and peak usage, as presented in Tables 4 and 5. This information helps identify whether an environment is prone to memory bottlenecks, especially under constrained configurations or when processing large batches.

Table 4. Memory Usage – CNN Model

Training Environment	Memory Usage	
	Start (%)	End (%)
Google Colaboratory	21.4	23
Kaggle Notebook	9	10.5

Table 5. Memory Usage – Decision Tree Model

Training Environment	Memory Usage	
	Start (%)	End (%)
Google Colaboratory	76.7	24
Kaggle Notebook	9.2	9.6

C. CPU Utilization

CPU utilization was tracked using the psutil. Python's cpu_percent() function captures usage before and after the training process to compute the average load exerted on the CPU in each environment. Tables 6 and 7 display the CPU utilization for the CNN and Decision Tree Models.

Table 6. CPU Utilization – CNN Model

Training Environment	CPU Utilization	
	Start (%)	**End (%)**
Google Colaboratory	50.2	94.9
Kaggle Notebook	11.2	93.5

Table 7. CPU Utilization – Decision Tree Model

Training Environment	CPU Utilization	
	Start (%)	**End (%)**
Google Colaboratory	38.5	63.2
Kaggle Notebook	2.7	27.1

4.5 Analysis and Comparison

This phase presents a comparative evaluation of training environments—Google Colaboratory and Kaggle Notebook—based on training time, CPU, and memory utilization for two models: a Convolutional Neural Network (CNN) and a Decision Tree.

A. Training Time Comparison

Kaggle Notebook significantly outperformed Google Colaboratory in training time for both models, as shown in Fig. 4. CNN training on Kaggle was approximately 62% faster, while Decision Tree training was about 38% faster, reflecting more efficient and stable resource utilization under free-tier conditions. Overall, Kaggle demonstrates superior training efficiency across both deep learning and traditional machine learning models, making it a more suitable platform for users with limited computational resources.

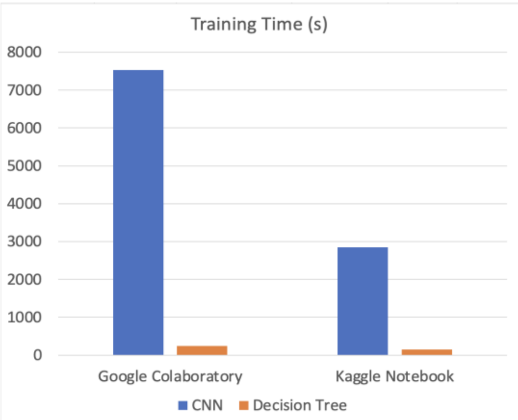

Fig. 4. Training Time (CNN and Decision Tree Model)

B. CPU and Memory Utilization

As shown in Fig. 5, Kaggle Notebook demonstrated lower and more stable memory usage than Google Colaboratory for both CNN and Decision Tree models. During CNN training, Kaggle used significantly less memory, while Google Colaboratory exhibited higher and more variable consumption. For the Decision Tree model, Google Colaboratory showed an initial memory spike, whereas Kaggle maintained consistently minimal memory usage. In terms of CPU utilization, Kaggle displayed a more efficient usage pattern, with a gradual ramp-up during CNN training and substantially lower CPU demand for Decision Tree training. These results indicate that Kaggle provides more efficient memory management and CPU scheduling under free-tier conditions.

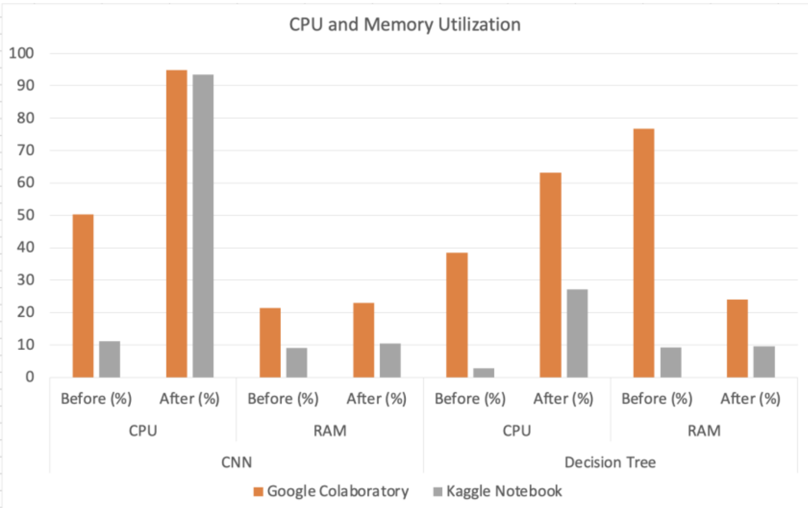

Fig. 5. Training CPU and Memory Utilization (CNN and Decision Tree Model)

4.6 Results Interpretation and Recommendation

A. Summary of Key Findings

This study compared Google Colaboratory and Kaggle Notebooks for machine learning training using CNN and Decision Tree models on the CIFAR-10 dataset. The results showed that Kaggle consistently outperformed Google Colaboratory, achieving 62% faster CNN training and 38% faster Decision Tree training, with lower and more stable memory usage. Although both platforms reached high peak CPU utilization, Kaggle exhibited a more gradual and efficient CPU load pattern, indicating better resource allocation. Overall, while both platforms successfully executed the models, Kaggle demonstrated superior training efficiency under free-tier conditions.

B. Interpretation of Comparative Results

Kaggle Notebook's superior performance is primarily due to its stable and predictable resource management. Unlike Google Colaboratory's dynamic allocation, Kaggle provides consistent training environments that lead to shorter training times, particularly for deep learning tasks. Its lower memory usage suggests more effective optimization mechanisms, while the gradual increase in CPU utilization reflects efficient scheduling and reduced resource contention. These differences emphasize the importance of platform-specific characteristics when selecting cloud-based free-tier environments for machine learning training.

C. Recommendations Based on Use Cases

Kaggle Notebook is the recommended platform for machine learning training under limited resources due to its faster training, efficient memory and CPU usage, and optimized scheduling for both deep learning and traditional models. However, Google Colaboratory remains suitable for educational and collaborative settings because of its Google Drive integration and ease of sharing. For individual researchers and competition-driven development, Kaggle offers a more stable and reproducible environment with minimal setup overhead.

D. Best Practice for Choosing a Training Environment

The choice of a cloud-based training environment should depend on model complexity, resource availability, and collaboration needs. For computationally intensive models such as CNNs, platforms like Kaggle are preferred due to reliable GPU access and lower memory usage. Benchmarking with smaller datasets, ensuring consistent software configurations, and fixing random seeds are essential for reproducibility and fair evaluation. Users should also consider session limits, particularly in Google Colaboratory, as Colab remains advantageous for projects that require frequent data sharing and collaborative work.

5 Conclusion

This study conducted a systematic performance analysis of two widely used cloud-based free-tier machine learning environments: Google Colaboratory and Kaggle Notebooks. By implementing and training machine learning models, specifically a Convolutional Neural Network (CNN) and a Decision Tree classifier, across both platforms using the CIFAR-10 dataset, the research aimed to assess and compare the training efficiency, memory consumption, and CPU utilization of each environment.

The results revealed that Kaggle Notebooks generally outperformed Google Colaboratory in terms of training speed, memory efficiency, and CPU resource management. Specifically, training time for the CNN model was reduced from 7,529.15 seconds on Colab to 2,843.95 seconds on Kaggle, representing a 62% improvement. Meanwhile, Decision Tree training decreased from 238.19 seconds on Colab to 146.83 seconds on Kaggle, resulting in a 38% improvement. Memory usage during CNN training on Kaggle was lower and more stable (9% to 10.5%) compared to Colab (21.4% to 23%), and Decision Tree training on Kaggle maintained memory usage between 9.2% and 9.6% compared to Colab's initial spike of 76.7%. CPU utilization patterns also favored Kaggle, particularly during Decision Tree training, where usage remained between 2.7% and 27.1%, while Colab showed a higher load, ranging from 38.5% to 63.2%.

These findings suggest that Kaggle Notebook is better suited for computational tasks under resource constraints, making it a more reliable platform for rapid prototyping, experimentation, and deep learning workloads under limited budgets. However, the study also recognizes that Google Colaboratory maintains advantages in terms of collaborative features and Google Drive integration, making it ideal for educational purposes and team-based development. Specific use cases, model complexity, and user needs should guide the choice between the two platforms.

This research contributes to the growing body of literature on cloud-based machine learning development by offering empirical evidence and recommendations for selecting appropriate cloud-based free-tier training environments. Future studies may extend this comparison to include additional platforms such as AWS SageMaker, Microsoft Azure, or paid tiers, as well as performance benchmarking across diverse datasets and model types.

References

[1] T. Carneiro, R. V. Medeiros Da NóBrega, T. Nepomuceno, G.-B. Bian, V. H. C. De Albuquerque, and P. P. R. Filho, "Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications," IEEE Access, vol. 6, pp. 61677–61685, 2018, doi: 10.1109/ACCESS.2018.2874767.

[2] A. Ali et al., "Enabling scalable and adaptive machine learning training via serverless computing on public cloud," Perform. Eval., vol. 167, p. 102451, Mar. 2025, doi: 10.1016/j.peva.2024.102451.

[3] P. Gimeno Sarroca and M. Sánchez-Artigas, "MLLess: Achieving cost efficiency in serverless machine learning training," J. Parallel Distrib. Comput., vol. 183, p. 104764, Jan. 2024, doi: 10.1016/j.jpdc.2023.104764.

[4] B. Cottier, R. Rahman, L. Fattorini, N. Maslej, T. Besiroglu, and D. Owen, "The rising costs of training frontier AI models," Feb. 07, 2025, arXiv: arXiv:2405.21015. doi: 10.48550/arXiv.2405.21015.

[5] D. Chahal, M. Mishra, S. Palepu, and R. Singhal, "Performance and Cost Comparison of Cloud Services for Deep Learning Workload," in Companion of the ACM/SPEC International Conference on Performance Engineering, in ICPE '21. New York, NY, USA: Association for Computing Machinery, Apr. 2021, pp. 49–55. doi: 10.1145/3447545.3451184.

[6] R. Pakdel and J. Herbert, "Adaptive Cost Efficient Framework for Cloud-Based Machine Learning," in 2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC), Jul. 2017, pp. 155–160. doi: 10.1109/COMPSAC.2017.42.

[7] E. Guerra, F. Wilhelmi, M. Miozzo, and P. Dini, "The Cost of Training Machine Learning Models Over Distributed Data Sources," IEEE Open J. Commun. Soc., vol. 4, pp. 1111–1126, 2023, doi: 10.1109/OJCOMS.2023.3274394.

[8] R. Teixeira, L. Almeida, M. Antunes, D. Gomes, and R. L. Aguiar, "Efficient training: Federated learning cost analysis," Big Data Res., vol. 40, p. 100510, May 2025, doi: 10.1016/j.bdr.2025.100510.

[9] S. M. Rajagopal, S. M., and R. Buyya, "Leveraging blockchain and federated learning in Edge-Fog-Cloud computing environments for intelligent decision-making with ECG data in IoT," J. Netw. Comput. Appl., vol. 233, p. 104037, Jan. 2025, doi: 10.1016/j.jnca.2024.104037.

[10] J. Lawrence, J. Malmsten, A. Rybka, D. A. Sabol, and K. Triplin, "Comparing TensorFlow Deep Learning Performance Using CPUs, GPUs, Local PCs and Cloud".

[11] P. Osypanka and P. Nawrocki, "Resource Usage Cost Optimization in Cloud Computing Using Machine Learning," IEEE Trans. Cloud Comput., vol. 10, no. 3, pp. 2079–2089, Jul. 2022, doi: 10.1109/TCC.2020.3015769.

[12] M. T. Islam, S. Karunasekera, and R. Buyya, “Performance and Cost-Efficient Spark Job Scheduling Based on Deep Reinforcement Learning in Cloud Computing Environments,” IEEE Trans. Parallel Distrib. Syst., vol. 33, no. 7, pp. 1695–1710, Jul. 2022, doi: 10.1109/TPDS.2021.3124670.

[13] T. Khan, W. Tian, G. Zhou, S. Ilager, M. Gong, and R. Buyya, “Machine learning (ML)-centric resource management in cloud computing: A review and future directions,” J. Netw. Comput. Appl., vol. 204, p. 103405, Aug. 2022, doi: 10.1016/j.jnca.2022.103405.

[14] G. Kasinathan and S. Jayakumar, “Cloud-Based Lung Tumor Detection and Stage Classification Using Deep Learning Techniques,” BioMed Res. Int., vol. 2022, no. 1, p. 4185835, 2022, doi: 10.1155/2022/4185835.

[15] V. Lahoura et al., “Cloud Computing-Based Framework for Breast Cancer Diagnosis Using Extreme Learning Machine,” Diagnostics, vol. 11, no. 2, Art. no. 2, Feb. 2021, doi: 10.3390/diagnostics11020241.

[16] G. Loukas, T. Vuong, R. Heartfield, G. Sakellari, Y. Yoon, and D. Gan, “Cloud-Based Cyber-Physical Intrusion Detection for Vehicles Using Deep Learning,” IEEE Access, vol. 6, pp. 3491–3508, 2018, doi: 10.1109/ACCESS.2017.2782159.

[17] C. Tong, X. Yin, S. Wang, and Z. Zheng, “A novel deep learning method for aircraft landing speed prediction based on cloud-based sensor data,” Future Gener. Comput. Syst., vol. 88, pp. 552–558, Nov. 2018, doi: 10.1016/j.future.2018.06.023.

[18] D. Wu, C. Jennings, J. Terpenny, and S. Kumara, “Cloud-based machine learning for predictive analytics: Tool wear prediction in milling,” in 2016 IEEE International Conference on Big Data (Big Data), Dec. 2016, pp. 2062–2069. doi: 10.1109/BigData.2016.7840831.

[19] S. Staravoitau, “Efficiency of Machine Learning Cloud-Based Services vs Traditional Methods in Stock Prices Prediction”.

[20] Á. López García et al., “A Cloud-Based Framework for Machine Learning Workloads and Applications,” IEEE Access, vol. 8, pp. 18681–18692, 2020, doi: 10.1109/ACCESS.2020.2964386.

[21] R. Pakdel, “Cloud-based Machine Learning Architecture for Big Data Analysis”.

[22] J. Sekar and A. Llc, “Deep Learning As A Service (Dlaas) In Cloud Computing: Performance And Scalability Analysis,” vol. 10, no. 3, 2023.

[23] G. P. Selvarajan, “Optimising Machine Learning Workflows In Snowflakedb: A Comprehensive Framework Scalable Cloud-Based Data Analytics,” vol. 8, no. 11, 2021.

[24] “Demo 2B: Google Colab for teaching CS and ML (SIGCSE TS 2024 - Demos) - SIGCSE TS 2024,” Sigcse.org, 2024. https://sigcse2024.sigcse.org/details/sigcse-ts-2024-demos/9/Demo-2B-Google-Colab-for-teaching-CS-and-ML (accessed Aug. 02, 2025).

[25] L. Quaranta, F. Calefato, and F. Lanubile, “KGTorrent: A Dataset of Python Jupyter Notebooks from Kaggle,” IEEE Xplore, May 01, 2021. https://ieeexplore.ieee.org/abstract/document/9463068/ (accessed Dec. 13, 2022).

[26] A. Felias, M. A. Remaldora, R. F. Vicente, and M. J. Timogan, “Forensic Facial Sketch Verification Using Image Processing Algorithms,” SMCC Higher Education Research Journal, vol. 3, no. 1, Jun. 2020, doi: https://doi.org/10.18868/ccs.03.060120.06.

[27] M. J. Timogan, “Implementation of Frequent Pattern (FP)-Growth Algorithm for Comparative Analysis on Teacher’s Performance Evaluation Results,” SMCC Higher Education Research Journal, vol. 3, no. 1, Jun. 2020, doi: https://doi.org/10.18868/cp-v3-a3.