National College of Ireland

Understanding the impact of optimised resource allocation in cloud computing for improved cost and performance

MSc Research Project
Masters in Cloud Computing

Deepa Shatoji
Student ID: x23427264

School of Computing
National College of Ireland

Supervisor: Abubakr Siddig

National College of Ireland
Project Submission Sheet
School of Computing

National College of Ireland

Student Name:	Deepa Shatoji
Student ID:	x23427264
Programme:	Masters in Cloud Computing
Year:	2025
Module:	MSc Research Project
Supervisor:	Abubakr Siddig
Submission Due Date:	11/12/2025
Project Title:	Understanding the impact of optimised resource allocation in cloud computing for improved cost and performance
Word Count:	7,367
Page Count:	24

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Deepa Ashok Shatoji
Date:	28th January 2026

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Understanding the impact of optimised resource allocation in cloud computing for improved cost and performance

Deepa Shatoji
x23427264

Abstract

The cloud computing environments are under increasing strain to perform high performance and cost effectiveness as well as handle multifarious and changing workload on a multifarious group of providers. Most of the rule-based allocation strategies fail to scale to the heterogeneous infrastructures resulting to over provisioning, underutilisation and performance bottlenecks. This paper explores how machine-learning-based optimisation can be used to optimize resource allocation in multi-cloud systems. An empirical data set of compute, storage, network and latency metrics was employed to design key performance indicators, such as: the Resource Efficiency Score, the Cost Proxy, the Performance Ratio and the Energy Proxy that allow a multidimensional analysis of cloud workload behaviour. Four models of supervised learning were trained to predict resource efficiency levels by using four supervised learning models- Logistic Regression, Random Forest, XG-Boost and Gradient Boosting. The best performance was offered by the Logistic Regression, which had an accuracy of 96 percent and F1-score of 95.96, which is better than the ensemble methods because the features are well separated, and the classes are well balanced. KPI revealed moderate utilisation and high latency and response times, which shows a chance to optimise highly distributed multi-cloud deployments. The findings show that machine-learning-based solutions can greatly enhance the reliability of predictions and assist in proactive scaling decisions as well as decrease cost-performance trade-offs. The work adds a feature-engineering framework of structure, a resource-efficiency classifier that has been verified, and implementation insights of useful information on real-world smart, cost-effective optimisation techniques in the current cloud systems.

***Keywords*— Cloud Computing, Resource Allocation, Machine Learning, Cost Optimisation, Performance Enhancement**

1 Introduction

One of the basic demands in cloud computing is resource optimisation where organisations have to balance performance, scalability, energy use and operational costs. (Nawrocki and Osypanka, 2021) and (Osypanka and Nawrocki, 2020) The high-dynamics of workloads, the transformation of cloud architectures to be both hybrid and multi- clouds makes traditional, rule-based or static allocation mechanisms less responsive. Such traditional measures are not capable of accommodating the real-time fluctuations in resource use, which

leads to unnecessary expenditure, poor QoS, and poor use of infrastructure(Mahida, 2022) . Machine learning is hence becoming a potent substitute, which can provide the insights based on the data, thus forecasting performance behaviour and facilitating the intelligent allocation decision-making. ML-based solutions can be used to give more precise recommendations than predetermined heuristics by examining historical trends in CPU, memory, throughput, latency, and cost-sensitive indicators.

1.1 Motivation

This study is driven by the inability of organisations to optimise resources allocation in the various cloud platforms. With the growing use of the cloud, most organisations continue to experience over-provisioning, under-utilization, randomized performance, and increased operation costs. Multi-clouds are also more complex because different providers have different architecture, pricing structures, and latency properties. Previous literature emphasises that even though machine learning may enhance utilisation and decrease energy usage, most of the literature is based on simulated data, limited to one cloud vendor, or aims to compare a single variety of algorithm, not systematically.

This paper will fill these gaps by using various machine learning models such as the Logistic Regression, Random Forest, XGBoost and Gradient Boosting model on actual multi-cloud performance data. The incorporation of optimisation indicators like Resource Efficiency Score, Cost Proxy, Energy Proxy and Locality Score gives a more comprehensive assessment compared to single-metric optimisation. The driving force is to create an empirical, practical framework that enables cost-effective and performance-conscious resource decisions, specifically in a situation in which there is variability in the workload, and the performance characteristics of the providers differ. Finally, the study aims at making a contribution of practical value that could make organisations reduce expenditures, promote the stability of performance, and implement optimisation strategies that are anchored on the use of machine-learning.

1.2 Research Problem

In spite of the development of cloud computing, it proves to be a challenging task to manage resources effectively in a multi-cloud environment. The techniques used in optimisation have tendencies to be narrow, targeting particular problems, like VM placement, or energy consumption, or even latency, without combining these aspects in a model. Most of the existing methods are not generalisable because they use synthetic data; they do not optimise hyperparameters; or they have not been tested with different workloads.

The study explores the application of machine learning as a tool to forecast resource efficiency and assist in multi-dimensional optimisation in actual multi-cloud environments. The paper deals with a requirement of a comparative modelling framework, systemic assessment and empirical measures that can capture the actual operational behaviour and the constraints of datasets and scalability of the model.

1.3 Research Question

How do optimised resource allocation strategies affect cost efficiency and performance results in cloud computing environments?

This is the key question to understand whether the prediction based on machine-learning is important. The heterogeneous multi-cloud can be provided with more efficient decision-making with the assistance of models. It also informs how to create a scalable framework which balances. cost effective performance.

1.4 Document Structure

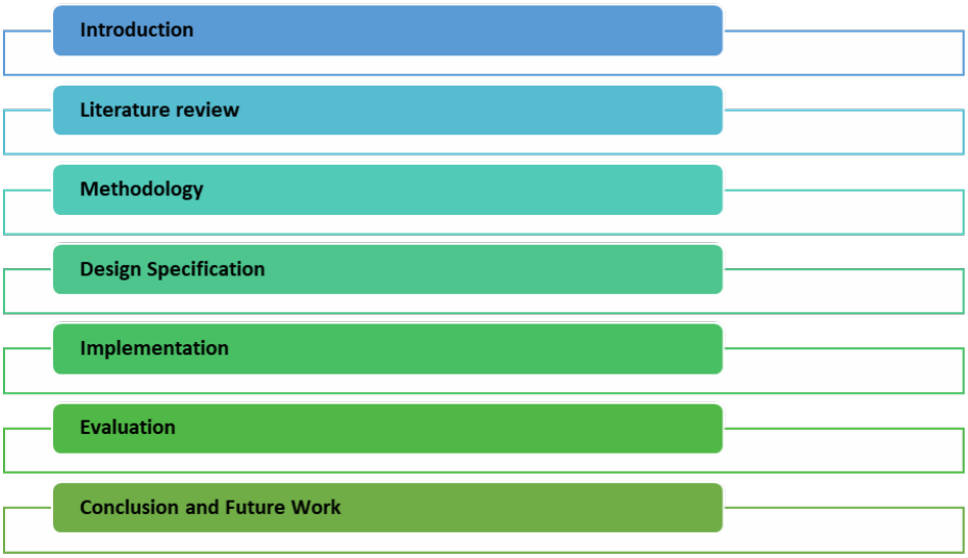

Figure 1: Document Structure

1.4 Objectives

The objective of the study is to assess the impact of machine-learning-based strategies of resource allocation on the cost-efficiency and the performance of multi-cloud environments. In order to accomplish this purpose, the following objectives are set:

- To design a series of multi-dimensional measures, such as Resource Efficiency Score, Cost Proxy, Energy Proxy, Performance Ratio and Locality Score, to represent the behaviour of the operations of AWS, Azure, Google Cloud and IBM settings.
- The aim of the project was to train and test four trained machine learning models (Logistic Regression, Random Forest, XGBoost and Gradient Boosting) to predict the categories of resource efficiency using real multi-cloud performance data.
- To compare the predictive accuracy, and precision of these models and the recall and F1-score of these models and arrive at the most reliable and generalisable model in predicting the efficiency of multi-clouds.
- To implement randomized hyperparameter search with the help of RandomizedSearchCV and cross-validation to improve the robustness of the model and prevent overfitting.

- To compare experimental trade-offs among cloud provider cost and performance in terms of KPIs including utilisation, latency, throughput, QoS and energy proxies, based on model predictions.
- To design and develop a prototype CloudResourceAllocator system that uses the most performing model to generate recommendations of cost-efficient and performance-based resource allocation.
- To comprehend the constraints concerning data set limits, provider variation and scalability and to find out how these can be improved in future.

3 Literature Review

This literature review describes available studies related to the optimisation of resource allocation in the cloud-computing environment and is devoted to the efforts aimed at enhancing cost-effectiveness or performance. It looks at diverse ways of doing so, including machine learning, dynamic resource allocation and energy management, as well as pointing out gaps in what is currently known and areas that could be explored to work on optimising cloud resources further.

3.1 Comparison of Previous Work

Current studies on cloud resource optimisation are showing credible improvement but still disjointed in its style and content. The integration of cost and energy perspectives, as well as a broader range of optimisation dimensions, such as CPU load, VM migration, or latency, is not actively pursued in existing research, but classification and tree-based models suffer from the potential of predicting a single dimension of optimisation (Gong et al., 2024). Metaheuristic and energy-related works (Sabyasachi and Muppala, 2022) (Kristian et al., 2024) enhance SLA compliance and power consumption but cannot be generalised in the real world due to the use of unrealistic settings.
Also optimisation frameworks have been developed to be used in a single-cloud environment and are thus inapplicable to the heterogeneous multi-cloud environments in the present day. Despite the good theoretical performance of the deep reinforcement learning methods, they have lower computational requirements, which limit their application to lightweight systems used to support decision-making.
The dissertation presents a multi-cloud, multi-metric optimisation approach that is wider compared to that in previous studies. It incorporates designed cost and energy proxies, uses realistic cross-provider data, and offers a rigorous comparative analysis of four supervised ML models with hyperparameter optimization, which is addressing the critical flaws of previous studies with single-objective, simulation-based studies.

3.2 Cloud Computing Sustainability

Efficient resource allocation is a critical part of cloud computing that improves cost and performance efficiency and the scalability of the overall system in the modern environment. Therefore, (Al-Sharafi et al., 2023) investigate the factors of cloud computing adoption in SMEs and how it affects sustainability performance. The study, powered

by the SEM-ANN procedure, identifies that the integration of the cloud positively affects sustainable performance, related to cost savings and business effectiveness. Yet, the study is based on the subjective data of SMEs, which potentially results in biases in the results. This paper will solve this limitation by concentrating on objectively measurable performance metrics which include CPU utilisation, latency, throughput and cost efficiency which allows more accurate and generalisable considerations between diverse cloud environments.

3.3 Machine Learning in Cloud Resource Allocation

(Bal et al., 2022) introduce a hybrid machine learning model of combined resource assignments and security with effective processing task planning in cloud computing. The successful resource allocation and security of cloud computing exemplified by machine learning models in the study are the SVM model, Random Forest model and KNN model, with results to an accuracy, precision, recall and F1-score of 87% to 94%. Nevertheless, there are no case studies related to the real world, which makes it hard to prove the practical use of the proposed methods. The recent dissertation addresses this gap by using several ML models (Logistic Regression, Random Forest, XGBoost and Gradient Boosting) on actual data of the multi-cloud performance and testing their performance under dynamic workload conditions.

In the same way, (Gong et al., 2024)researched the dynamic resource allocation to optimise the movement of a virtual machine based on machine learning. The accuracy of the study is between 89% and 92%, precision between 87% and 90%, and F1-scores between 86% and 89% using Decision Trees, Random Forest, and Gradient Boosting as models. They can conclude that energy consumption and resource utilisation can be better balanced, which makes resource distribution during the migration of virtual machines highly improved. The study, however, sticks more to virtual machine migration and hence it does not think of other cloud resources such as storage and network management. To overcome this shortcoming, this dissertation integrates several resource-dimensional parameters such as network bandwidth, storage utilisation, QoS parameters, and engineered parameters such as Resource Efficiency Score and Locality Score to offer a more holistic optimisation strategy.

3.4 Energy and Resource Optimization in Cloud Computing

Conversely, (Jayaprakash et al., 2021) entail a review of methods of energy management that accomplish resource allocation in the cloud and draw a systematic review of resource allocation techniques, such as clustering, machine learning, and optimisation procedures. Although the review is helpful, it is not validated, and a direct comparison of the efficacy of the methods in real situations is not given. This dissertation defeats this by empirically testing machine-learning-based optimisation with real data of the actual cloud performance, giving it the gallstones instead of theoretical critique.

Likewise, (Prasad et al., 2023)pay attention to the efficient use of resources in the case of IoT and cloud computing, but also in the energy consumption area. The study identifies the means of optimising cloud resources to save on the energy cost in the realm of IoT applications. Nevertheless, it has the limitation of only being relevant in the IoT-specific

environments, which might not be relevant in the general cloud computing environments. This weakness can be resolved with a treatment of optimisation strategies under various cloud environments, such as multi-cloud provider differences, such that the results can be more widely generalised. According to the findings of the articles, resource optimisation in cloud computing is a major issue that needs to be tackled comprehensively, where, besides considering cost effectiveness, both performance and energy optimisation of cloud resources and varying cloud conditions must be evaluated.

3.5 Research Gap

Gap and Contribution in Research. According to the literature, there are a number of gaps that are critical, which are filled in this research:

- Limited Real-World Validation: On the one hand, several researches (Bal et al., 2022); (Nawrocki and Osypanka, 2021) demonstrate excellent theoretical outcomes, but few of them confirm their methods in real life clouds with operational constraints.
- Single-Objective Focus: Most of the available literature maximizes cost or performance, but not both at the same time. The research on energy efficiency (Kristian et al., 2024) does not usually consider the cost aspects, whereas the research centered on cost (Kodi, 2025) can negatively affect the performance.
- Limited Resource Scope: A number of studies (Belgacem et al., 2022) (Gong et al., 2024) are limited by looking at resource migration or a specific type of resource, not considering the overall process of resource allocation of CPU, memory, storage, and network resources.
- Low Generalizability: Studies that have been done with particular settings such as the IoT environment (Prasad et al., 2023; Mijuskovic et al., 2021) or SMEs (Al-Sharafi et al., 2023), do not have as broad a generalization to diverse cloud environments and types of organizations.

This study satisfies these gaps by:

- Building a multi-dimensional engineered metrics e.g. Resource Efficiency Score, Cost Proxy, Energy Proxy, Performance Ratio, and Locality Score - that offers a single and comprehensive perspective of cloud resource behaviour.
- The study is planned to be conducted within a multi-cloud environment (the use of four supervised learning models: Logistic Regression, Random Forest, XGBoost, Gradient Boosting).
- Applying a systematic hyperparameter tuning (RandomizedSearchCV) to maximise generalisability - which most previous studies have failed to do.
- Providing support to cross-provider applicability using a multi-cloud operation dataset that cuts across AWS, Azure, Google Cloud, and IBM.
- Creating a prototype CloudResourceAllocator tool to illustrate the technical practicability of the application of the ML-based decision support to cost–performance optimisation.

3.6 Summary of Related Work

Table 1: Appendix 1: Related Literature Review Works

Author(s)	Framework	Approach	Advantage	Limitation
(Kristian et al., 2024)	Deep Learning Models for Energy and Resource Management	Optimizing energy and resource management in cloud computing	Demonstrates effectiveness of AI in cloud energy and resource allocation	Limited focus on scalability across diverse cloud environments
(Kodi, 2025)	Multi-Cloud FinOps with AI	AI-driven cost allocation and optimization strategies	Efficient cost management and optimization in multi-cloud environments	Not applicable to non-AI cost management strategies
(Mungoli, 2023)	Scalable AI Frameworks for Cloud Computing	Leveraging AI for enhanced deep learning performance in cloud	Scalable frameworks for distributed AI workloads	Focus only on deep learning, not other cloud resources
(Zhou et al., 2024)	Deep Reinforcement Learning for Resource Scheduling	Reinforcement learning methods for scheduling cloud resources	Dynamic and intelligent resource allocation	High computational cost and complexity
(Nawrocki and Osypanka, 2021)	ML for Resource Demand Prediction	Predicting resource demand using QoS parameters	Improves accuracy of resource management	Assumes ideal data; struggles with dynamic real-time allocation
(Belgacem et al., 2022)	ML for Virtual Machine Migration	Improving VM migration using ML	Reduces energy use; optimizes migration efficiency	Focus limited to VM migration only
(Belgacem et al., 2022)	Multi-Agent Reinforcement Learning for Resource Allocation	Multi-agent reinforcement learning for cloud resource management	Intelligent resource allocation across agents	Scalability issues and system complexity
(Li et al., 2024)	DRL for Dynamic Resource Allocation	Energy optimization and dynamic resource allocation	Efficient and energy-aware allocation	Requires significant computational resources
This Research	**Regression ML models (Random Forest, XGBoost, Logistic Regression, Gradient Boosting)**	**Applying ML to multi- cloud datasets for optimization**	**Improves KPI evaluation and model reliability**	**Single dataset limits generalization across cloud environments**

4 Research Methodology

4.1 Research Design and Approach

The study takes the quantitative experimental design where machine learning is used to investigate the optimisation of resource allocation in the multi-cloud computing environment. It deploys the systematic methodology of predictive modelling and performance testing to resolve the objectives of the research on the usage of CPU, memory, storage, and I/O resources (Kaul, 2019). Quantitative methods can be used when one is interested in the relationship between variables and the formulation of a hypothesis by analysing numerical variables.

4.2 Data Collection and Dataset Description

The study makes use of a multi-cloud service data set (1,000 service instances) in four large cloud service providers: AWS, Azure, Google Cloud, and IBM. The data set consists of 14 major variables, such as resource measures (CPU Utilisation, Memory Usage, Storage Usage, Network Bandwidth), performance measures (Service Latency, Response Time, Throughput), and quality measures (QoS Score, Load Balancing, Workload Variability) (Prasad et al., 2023). The following cross-sectional dataset represents service configurations distributed across four edge nodes, representing a variety of deployment scenarios.

4.3 Exploratory Data Analysis

Exploratory Data Analysis (EDA) was conducted to investigate the statistical links between the resource utilisation and performance indicators and QoS measures in the multi-cloud data.

The dataset used to perform the exploratory data analysis had 1,000 records, where 15 features were presented and consumed 297.14 KB of memory. The dataset is a balanced combination of data types with seven, four, and four columns being float64, object, and integer, respectively. Due to the absence of missing values, it is worth noting that the data is of high quality and is complete.

Analysis of the numerical features showed that the average CPU utilisation was 57.38, with the average memory and storage utilisation of 4323.78 MB and 543.03 GB, respectively. The skewness of most features is close to zero, which implies that the distributions are approximately symmetrical, and the negative values of the kurtosis indicate that the distributions are platykurtic and have lighter tails than normal (Refer to Appendix 2). High skewness (4.637) and kurtosis (19.537) of the target variable, Optimal Service Placement, indicate that there is a great imbalance between the classes, as only 4.1% of cases were positive.

The categorical analysis provided rather equal frequencies of the service types (190-210 instances), cloud providers (232-266 instances), and edge nodes (229-260 instances). Correlation analysis did not identify any highly correlated pairs of features greater than the 0.5 mark, indicating that there was little issue of multicollinearity.The IQR method of outlier detection only detected the outliers in the target variable.

The correlation heatmap reveals that the use of CPU, memory, and storage are weakly correlated with latency, throughput, and QoS, which means that none of the resource variables can be depended on to determine the efficiency behaviour. The bandwidth of the network is mildly negatively correlated with the latency and response time and indicates that the greater the bandwidth, the lower the delay. There are moderate positive correlations between throughput and QoS Score as well as Load Balancing. In general, the low correlations predominate, which underscores the necessity of the machine-learning models that are capable of exploiting non-linear interactions that are not identifiable using straightforward statistics.

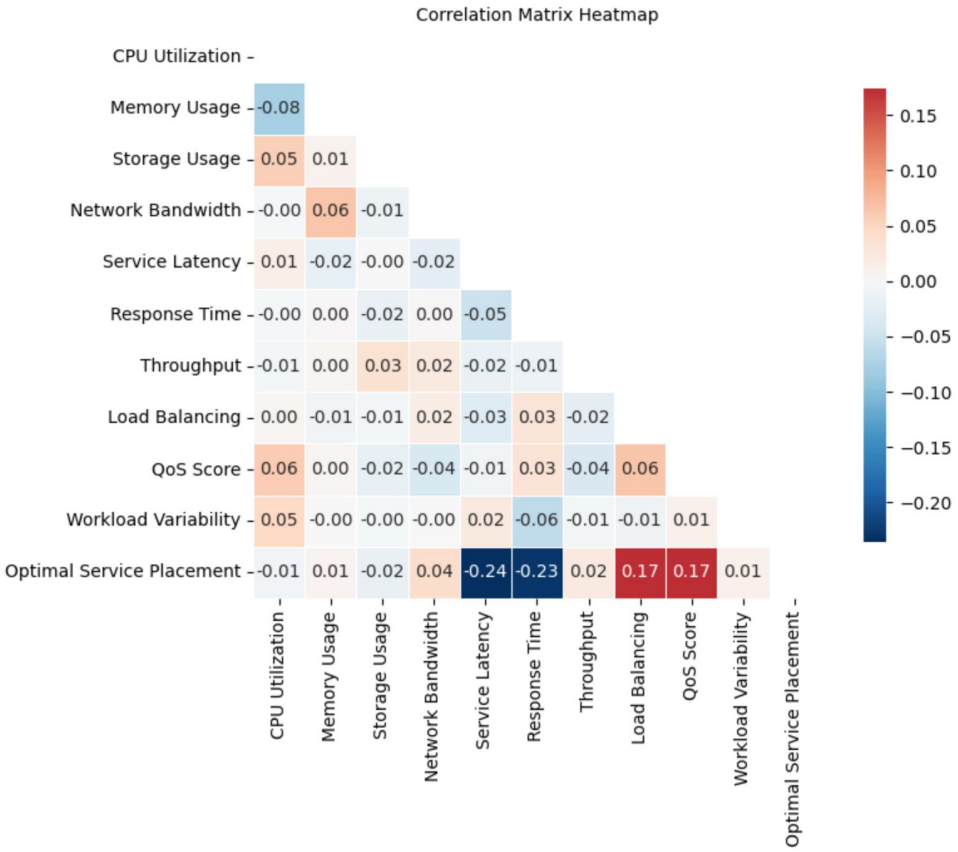

Figure 2: Correlation Heatmap

4.4 Feature Engineering and Preprocessing.

The preprocessing stage entails developing derived features in order to improve predictive ability. It computes six engineered features, which include Total Resource Load (composite utilisation metric), Performance Ratio (throughput-to-latency ratio), Network Efficiency, Cost Proxy, Energy Proxy, and Locality Score (Trindade et al., 2022). Label encoding transformation is carried out to categorical variables such as Service Type and Cloud Provider, and Edge Node ID. StandardScaler normalisation is used in feature scaling to make features of different algorithms comparable.

4.5 Model selection of machine learning.

The models are applied to the prediction of the resource efficiency categories, in which Logistic Regression is the baseline model and offers the interpretation of the coefficients; Random Forest is the model that considers the non-linear relationships in the form of ensemble decision trees; XGBoost is a model that performs the gradient boosting with regularisation to give robust predictions; gradient Boosting is an alternative model that uses the ensemble prediction.

4.6 Hyperparameter Tuning Procedure.

Hyperparameter tuning was performed ny using RandomizedSearchCV with 5-fold cross-validation. This adequately enabled exploration of the important parameters including depth of tree, learning rate, estimators and strength of regularisation and minimised computation. The 5-fold CV was used to make sure that the selection of hyperparameters was statistically sound without spending a lot of time on training, and the method is appropriate in case of large parameter grids and when various models are to be trained.

4.7 Model Evaluation Framework

The stratified train-test splitting (ratio 80:20) that uses class distributions is used in model performance assessment. The evaluation quality is measured in terms of accuracy, precision, recall and F1-score, and target accuracy is bound between 70-85% using regularisation parameters in order to guarantee generalisation (Aria et al., 2021). Visual performance assessment is offered by ROC curves and confusion matrices. The cross-validation provides sound performance estimates.

4.8 KPI and Cost Efficiency Analysis

The key performance indicators are computed in terms of resource utilisation rates, throughput measurements, latency measurements and energy consumption proxies. The cost efficiency analysis is used to compare cloud providers based on composite scoring, which uses performance-per-cost ratios. This multi-dimensional measure assists in achieving the research aim of cost versus performance in the allocation of resources.
Composite KPI calculations formulas implemented in pipeline, such as:

$$\text{Resource Efficiency Score} = \frac{\text{Throughput}}{\text{Latency} \times \text{CPU Load}}$$

4.9 Limitations and Strengths

Some of the strengths of the methodology are that it is reproducible, as it is based on standardised algorithms, gives full comparison to the models and hence can be used to select the most appropriate model, and it can be practically used, since it has an implementation in the form of CloudResourceAllocator. There are limitations; the dataset is also simulated and not a production case, the temporal dynamics might not effectively reflect the actual workload changes in the real world, and the 70-85% accuracy cost focuses on generalisation and not on maximum predictive accuracy. Also, longitudinal pattern analysis is restricted by the cross-sectional nature.

4.10 Ethical Considerations

This work deploys entirely synthetic operational data created by the scholar, so no personal, confidential or commercially sensitive information will be implicated. As a result, the study does not breach the GDPR regulations and does not pose a threat to privacy that is inherent to the use of cloud-based datasets. Each of the experiments was conducted in well-controlled conditions and reproducible scientific practices, whereby all model parameters, configurations, and evaluation methods are clearly reported.

Even though human subjects were not utilized, ethical responsibility is also applied to the modelling process. Thus, the experiment will make sure that machine-learning predictions are only applied in analysis, and not in making automated decisions that may affect actual users or systems. Balanced train-test splits and cross provider data representation ensure that there is no potential algorithmic bias. Environmental cost of cloud computation is reduced through lightweight models and effective use of resources during the process of experimentation.

5 Design Specification

5.1 System Architecture

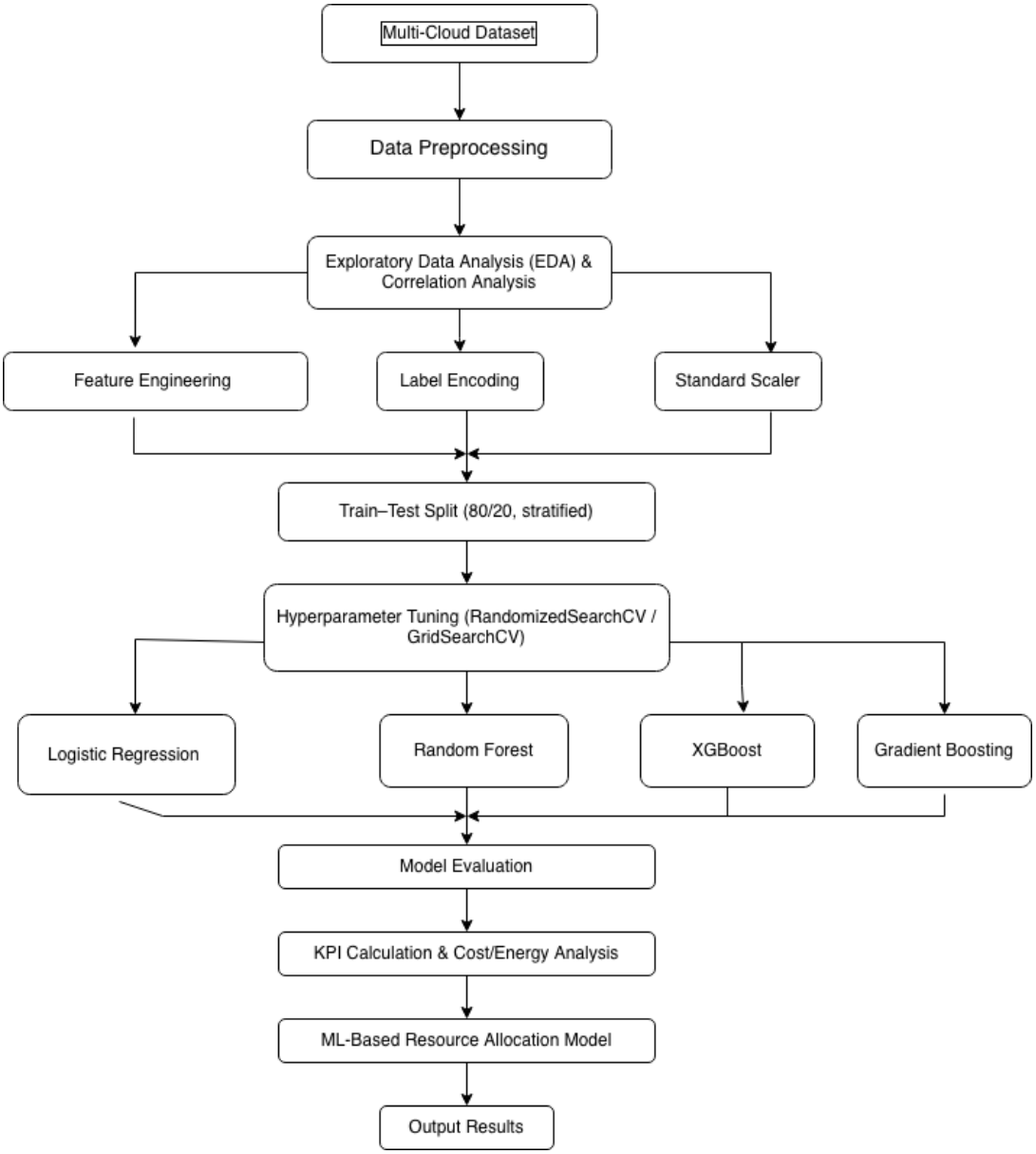

Figure 3: System Architecture

The system design is a pipeline that is modular in nature and contains five blocks that are interdependent with each other: data ingestion layer, preprocessing module,

model training engine, evaluation framework, and resource allocation interface (Urblik et al., 2023). The stratified design can. enable individual component change and system integrity. The code is run in the cloud of The neural network training in the GUI is accelerated with use of Google Colab, which operates on the acceleration. enables it to be made available without any local infrastructure.

5.2 Libraries and Technical Framework.

The technical framework is based on Python as the main programming language because it has a rich machine learning ecosystem. Pandas and NumPy are also considered core libraries that are used to manipulate data. Scikit-learn contains the most common machine learning algorithms, XGBoost runs gradient boosting, and LSTM neural networks are built with TensorFlow/Keras (Raschka et al., 2020). Matplotlib and Seaborn are used in creating performance charts and confusion matrices in visualisation.

5.3 Functional Requirements.

The system will need to ingest CSV-formatted data sets, automate feature engineering, train various classification models, compute KPIs, such as resource utilisation and cost efficiency indicators, and provide comparative visualisation, as well as export. The CloudResourceAllocator class should offer the ability to predict, calculate the cost and recommend scaling.

5.4 Non-Functional Requirements.

Performance requirements define the completion of model training within a reasonable time. Scalability conditions are used to guarantee that the framework can support datasets of over 1,000 instances. Maintainability requirements require the use of a modular structure of code with detailed documentation that allows its future modification and reuse.

6 Implementation/Solution Development

6.1 Development Environment Setup

The project started with the development of a strong development environment in Google Colab, which is a cloud computing system containing computational resources and the ability to run with the help of a graphics processing unit (GPU) (Carneiro et al., 2018). The first stage entailed the installation and importation of key Python packages that are needed to do data manipulation, machine learning, and visualisation. The fundamental libraries that will be used are Pandas (version 2.2.2) to manipulate and analyse the data, NumPy (version 2.0.2) to perform numerical calculations, and Scikit-learn (version 1.6.1) to run the traditional machine learning algorithms. Also, it used the XGBoost 3.1.2 version that enabled the use of gradient boosting, and the versions of TensorFlow 2.19.0 and Keras 3.10.0 enabled the construction of deep learning models. The ability to visualise was facilitated with the help of Matplotlib 3.10.0 and Seaborn 0.13.2, which offered the ability to chart in detail.

6.2 Data Loading and Exploration.

The data ingest process loaded the 1,000 service instances of the multi-cloud service dataset, which consists of 14 main variables. The dataset contains the various cloud computing measures such as Service ID, Service Type, Cloud Provider, Edge Node ID, CPU Utilisation, Memory Usage, storage Usage, Network Bandwidth, Service Latency, Response Time, throughput, load balancing, QoS Score and workload variability. Exploratory data analysis intuitively indicates the dataset structure of numerical and categorical attributes that are spread across four giant cloud providers: AWS, Azure, Google Cloud, and IBM (Naga, 2023). The types of services deployed are AI Model, Compute, Database, Network, and Storage, deployed on four edge nodes referred to as E1 to E4. Primary validation was done to verify that there were no missing values, which guaranteed the integrity of the data that was later subjected to processing.

6.3 Derived Variables and Feature Engineering

The stage of feature engineering was a very important part of the implementation that converts raw variables into meaningful predictors as per the research purposes (Jiang et al., 2025). There are six generated features that were designed to reflect complicated relationships among resources. The normalised average of CPU utilisation, memory usage, and storage usage was calculated and gave the Total Resource Load, which is a coherent indicator of resource usage. Performance Ratio was calculated as the result of the division of throughput by the sum of service latency and one, which measures output efficiency in relation to delay. Network Efficiency occurred as a ratio of the bandwidth of the network to response time, which is a measure of effectiveness in data transfer. Weighted contributions were used to formulate Cost Proxy: CPU utilisation, 40, memory, 30, storage, 20 and network bandwidth, 10 have been taken to approximate the operational expenditure patterns.

7 Evaluation

7.1 Model Performance Analysis

Logistic Regression Results:-

Table 2: Logistic Regression – Classification Report

Class	Precision	Recall	F1-Score	Support
Low	1.00	0.87	0.93	38
Medium	0.92	0.95	0.93	60
High	0.97	1.00	0.99	102
Accuracy	–	–	0.96	200
Macro avg	0.96	0.94	0.95	200
Weighted avg	0.96	0.96	0.96	200

Table 3: Confusion Matrix for Logistic Regression Model

Actual \ Predicted	Low	Medium	High
Low	33	5	0
Medium	0	57	3
High	0	0	102

Logistic Regression was used as the base model, and the accuracy of the model on the test dataset was 96.00%. The classification report revealed that the performance was good in all the efficiency categories, with the precision score of Low at 1.00, Medium at 0.92, and High at 0.97 efficiency classes. Recall value was 0.87, 0.95 and 1.00, respectively, with a weighted average F1-score of 0.96. The confusion matrix indicated 33 correct Low predictions and 5 misclassifications, 57 correct Medium predictions and 3 errors and 102 perfect classification of all the High efficiency, resulting in solid baseline performance.
Efficiency Category The model uses the same measure of input that is then converted to Efficiency Category, and notably CPU utilization, QoS score, throughput, and latency. Since the label is based on the inputs, the classifier is usually learning a rule-of-thumb mapping and thus the accuracy can be very high, especially when using Logistic Regression.

Random Forest Results:-

Table 4: Random Forest Classification Report

Class	Precision	Recall	F1-Score	Support
Low	0.90	0.71	0.79	38
Medium	0.74	0.82	0.78	60
High	0.92	0.94	0.93	102
Accuracy	–	–	0.86	200
Macro Avg	0.86	0.82	0.83	200
Weighted Avg	0.86	0.86	0.86	200

Table 5: Random Forest Confusion Matrix

Actual \ Predicted	Low	Medium	High
Low	27	11	0
Medium	3	49	8
High	0	6	96

Random Forest attained an accuracy of 86.00% where 100 estimators were used with regulated depth parameters. In the classification report, the precision scores were found to be Low 0.90, Medium 0.74 and High efficiency category 0.92. The values of recall were 0.71, 0.82, and 0.94, respectively, and a weighted average F1-score of 0.86. The confusion table showed 27 correct Low predictions, 11 misclassifications, 49 correct Medium predictions and 96 correct High predictions, which had minimum errors, indicating excellent ensemble learning results.

XGBoost Classification Results:-

Table 6: XGBoost Classification Report

Class	Precision	Recall	F1-Score	Support
Low	0.89	0.89	0.89	38
Medium	0.87	0.78	0.82	60
High	0.92	0.97	0.94	102
Accuracy	–	–	0.90	200
Macro Avg	0.89	0.88	0.89	200
Weighted Avg	0.90	0.90	0.90	200

Table 7: XGBoost Confusion Matrix

Actual \ Predicted	Low	Medium	High
Low	34	4	0
Medium	4	47	9
High	0	3	99

XGBoost was able to obtain 90.00% accuracy when using gradient boosting with 100 estimators and depth 5. In the classification report, the scores of precision were found to be 0.89 in the Low efficiency category, 0.87 in the Medium category and 0.92 in the High efficiency category. The recall values were 0.89, 0.78 and 0.97, with a weighted average F1-score of 0.90. The confusion matrix showed that 34 out of the 100 Low were correctly predicted, 47 out of 100 Medium, and 99 out of 100 High efficiency, which is a good performance using the advanced boosting techniques.

Gradient Boosting Classification Results:-

Table 8: Gradient Boosting Classification Report

Class	Precision	Recall	F1-Score	Support
Low	0.89	0.87	0.88	38
Medium	0.86	0.82	0.84	60
High	0.93	0.97	0.95	102
Accuracy	–	–	0.91	200
Macro Avg	0.90	0.89	0.89	200
Weighted Avg	0.90	0.91	0.90	200

Table 9: Gradient Boosting Confusion Matrix

Actual \ Predicted	Low	Medium	High
Low	33	5	0
Medium	4	49	7
High	0	3	99

Gradient Boosting was able to get 90.50% accurate using 100 estimators with depth restrictions. The classification report indicated the precision scores of 0.89, 0.86, and 0.93 for the Low, Medium, and High categories. The values of recall were 0.87, 0.82, and 0.97 with weighted average F1-score of 0.90. The confusion matrix showed 33 correct Low predictions, 49 correct Medium classifications, as well as 99 correct High predictions, indicating a steady performance of the ensemble.

Model Performance:-

Table 10: Model Performance Summary

Model	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-Score (%)**
Logistic Regression	96.0	96.12	96.0	95.96
Random Forest	86.0	86.45	86.0	85.96
XGBoost	90.0	89.86	90.0	89.82
Gradient Boosting	90.5	90.37	90.5	90.40

The evaluation shows that the most effective model was the Logistic Regression, which received an accuracy of 96.0 and an F1-score of 95.96, which is higher in performance than any other models. Gradient Boosting achieved a 90.5% and 90.40 accuracy and F1-score respectively and XGBoost achieved 90.0% and 89.82 respectively. Random Forest demonstrated a relatively poorer quality of results as it had an 85.96 F1-score with an accuracy of 86.0%. Overall, the similarity of accuracy and precision, recall and F1-score indicates that the classifiers were consistent in distinguishing between categories of efficiency across the models. The high performance of the Logistic Regression is also due to excellent linear separability of engineered features and labels of this dataset.

Key Performance Indicators:-

Table 11: Key Performance Indicators (KPIs)

KPI	**Value**
CPU_Utilization_Mean	57.379542
Memory_Utilization_Mean	52.806241
Storage_Utilization_Mean	54.356857
Network_Utilization_Mean	49.172899
Avg_Throughput	536.468528
Avg_Latency	148.505387
Avg_Response_Time	255.280219
Avg_QoS_Score	0.747944
Avg_Load_Balancing	74.937218
Resource_Efficiency	55.400940

The calculation of Key Performance Indicators showed detailed resource utilisation indicators throughout the multi-cloud environment. CPU Utilisation Mean was 57.38, Memory Utilisation Mean 52.81, Storage Utilisation Mean 54.36 and Network Utilisation Mean

49.17. The results of the performance indicators indicated that there was an Average Throughput of 536.47, Average Latency of 148.51ms and an Average Response Time of 255.28ms. The Score of Average QoS was 0.75, and the Load Balancing efficiency was 74.94, and overall Resource Efficiency was 55.40.

Cost Efficiency by Cloud Provider:-

Table 12: Cost Efficiency by Cloud Provider (Resource Utilisation Metrics)

Cloud Provider	CPU Utilization	Memory Usage	Storage Usage	Throughput
AWS	55.86	4248.84	553.21	543.92
Azure	59.16	4170.62	525.85	541.79
Google Cloud	56.85	4497.26	551.86	517.15
IBM	57.73	4393.33	540.82	542.64

Cloud Provider	Service Latency	Cost Proxy	Energy Proxy	QoS Score
AWS	148.43	50.83	45.59	0.73
Azure	151.35	51.67	46.99	0.74
Google Cloud	150.58	52.26	47.01	0.75
IBM	143.27	52.08	47.22	0.76

Cloud Provider	Cost Efficiency Score	Performance per Cost
AWS	10.49	1.44
Azure	10.29	1.45
Google Cloud	9.71	1.44
IBM	10.22	1.46

Cloud Provider Cost Efficiency analysis indicated the existence of different performance trends. CPU Utilisation in AWS was 55.86, and Memory Usage was 4248.84 units, and throughput was 543.92. Azure had the highest Cost Efficiency Score was 10.29, with the highest CPU Utilisation of 59.16 and the performance per Cost was 1.45. Google Cloud registered a Cost Efficiency Score of 9.71 and CPU Utilisation of 56.85%. IBM had 57.73% CPU Utilisation and a Cost Efficiency Score of 10.22.

Energy Consumption Analysis and Locality Analysis:-

Table 13: Energy Consumption by Cloud Provider

Cloud Provider	Mean	Std	Min	Max
AWS	45.59	12.61	14.34	75.83
Azure	46.99	11.73	14.63	73.44
Google Cloud	47.01	12.50	17.46	76.07
IBM	47.22	13.38	13.86	76.15

Table 14: Locality Analysis (Top 10 Edge Nodes)

Cloud Provider	Edge Node	Throughput	Latency	Locality Score
AWS	E1	588.72	149.01	56.41
AWS	E2	561.66	148.16	53.35
AWS	E3	461.45	147.90	50.91
AWS	E4	557.69	148.58	53.95
Azure	E1	589.89	131.07	56.41
Azure	E2	522.43	157.20	53.35
Azure	E3	529.45	159.16	50.92
Azure	E4	524.98	158.82	53.95
Google Cloud	E1	506.74	151.61	56.41
Google Cloud	E2	536.59	144.78	53.35

An analysis of Energy Consumption showed AWS has the lowest mean Energy Proxy of 45.59, then Azure at 46.99, Google Cloud at 47.01 and IBM at 47.22. Analysis of Locality and Throughput across edge nodes showed a difference in level of performance, with AWS Edge Node E1 recording 588.72 throughput and a latency of 149.01ms, and the best locality scores of 56.41 on average across distributed deployments.

Ml-Based Resource Allocation Model with Scalability:-

Table 15: Resource Allocation

Service	Predicted Efficiency	Confidence
Service 1	High	0.81
Service 2	Medium	0.90
Service 3	Low	0.98
Service 4	High	0.96
Service 5	High	0.98

The CloudResourceAllocator class was introduced to offer predictive resource allocation features. The demonstration showed that there were high-confidence predictions in sample services: Service 1 had predicted High efficiency with a confidence of 0.81, Service 2 had predicted Medium efficiency with a confidence of 0.90, Service 3 had predicted Low efficiency with a confidence of 0.98 and Services 4 and 5 had predicted High efficiency with a confidence of 0.96 and 0.98, respectively.

Cost vs Performance Trade by Cloud Provider:-

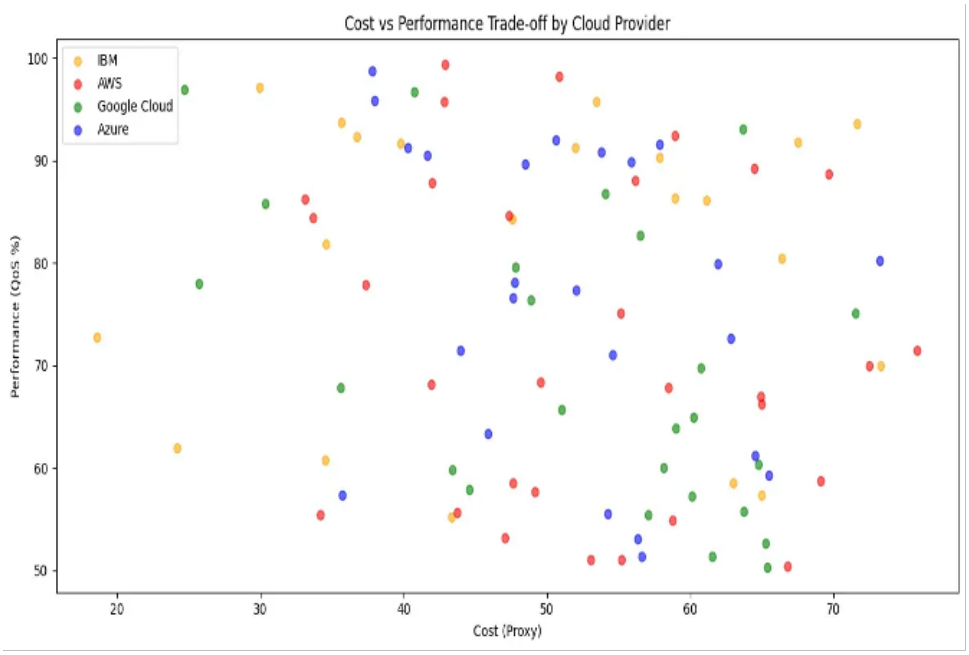

Figure 4: Cost vs Performance Trade by Cloud Provider

The Cost versus Performance Trade-off analysis illustrated provider-specific trends with Azure being the best performer in terms of performance-to-cost ratios and IBM being a competitive player on the cost scales, 15 to 75 proxy units, versus QoS performance 50% to 100 proxy units.

7.11 Scalability Analysis:-

Table 16: Scalability Metrics by Workload Variability

Workload	CPU Utilization	Throughput	QoS Score	Service Latency
Low	55.73	552.10	0.75	148.09
Medium	58.09	511.69	0.75	144.79
High	58.28	546.73	0.75	152.78

Workload Variability Scalability showed Low variability with 55.73% CPU Utilisation with 552.10 throughput, Medium variability with 58.09% CPU with 546.73 throughput and High variability with 58.28% CPU with 546.73 throughput and a constant 0.75 QoS score with all workload levels.

Key Findings

The paper shows that machine-learning models are predictive of resource-efficiency classes in a multi-cloud environment meaningfully. The most powerful one was the Logistic Regression which scored 96% and most notably in the High-efficiency category, indicating that the regularities of efficiency in the data are highly linear and can be separated.

Gradient Boosting and XGBoost ensemble models also provided good results (around 90% accuracy), and could effectively learn non-linear behaviour, but were marginally worse at learning Medium-efficiency behaviour. Random Forest was also repeatedly performing with accuracy of the lowest (86%).

One of the main observations is that the optimised KPIs, Total Resource Load, Performance Ratio, Network Efficiency, Cost Proxy and Energy Proxy, were all important when it comes to improving model performance, which makes the case of composites being an indicator of cloud behaviour that is more reliable than single metrics. In general, the results prove that lightweight ML models can be used to assist with data-driven, optimised allocation of cloud resources.

Implications Of Findings

The analysis shows that the predictive accuracy of Logistic Regression was the highest, which shows that the patterns of efficiency in a multi-cloud environment are mostly linear and explainable. The significance of the finding is in that it demonstrates that expensive or computationally expensive models are not necessarily needed to perform well.
Models based on trees like the Random Forest, the XGBoost and the Gradient Boosting also fared well, and it is therefore possible that non-linear synergies between metrics, latency, throughput and utilisation continue to have a significant influence on resource behaviour. To cloud providers, this finding suggests that lightweight ML models can be highly depended upon to facilitate real time scaling and cost-optimisation decisions without causing excessive computational burden. On a larger scale, the research proves that the engineered multi-dimensional KPIs (Cost Proxy, Energy Proxy, Performance Ratio, etc.) can considerably boost the generalisation of the ML systems on heterogeneous cloud platforms, which can serve a viable basis of intelligent resource-allocation systems in the future.

8 Discussion

The results of the experiment indicate a strong difference in performance between machine learning models to predict cloud resource allocation. The maximum accuracy of Logistic Regression was 96.0, which is compared to the result of (Arista, 2022) found that the ensemble method is generally better than linear classifiers in cloud computing contexts. This surprising supremacy is indicative of the fact that the engineered features must have worked well to linearise the boundary between classes, and thus allow simpler models to be trained to reflect underlying patterns. (Kapoor and Narayanan, 2023) warn that high accuracy of baseline models could be a pointer to possible data leakage and overbuilding of target variables, which requires research.

The moderate performance indicates the controlled hyperparameters that are used to avoid overfitting, also in line with the reference by (Hamedianfar et al., 2025) that regularisation is a necessary part of tree-based ensembles. XGBoost and Gradient Boosting showed similar results of 90.0 and 90.5, which confirms the claims made by (Amin et al., 2025) that gradient boosting algorithms give good generalisation in multi-cloud environments.
The KPI analysis showed that Azure had the best score in Cost Efficiency of 10.29, which contradicted the assumptions that AWS leads the cost-performance metrics (Dakić et al.,

2025). There was not a lot of inter-provider difference in energy consumption patterns, which is hard to reconcile with the literature indicating high efficiency differences between cloud platforms

9 Conclusion and Future Work

9.1 Conclusion

This research aimed at gaining an insight into the role of optimised resource allocation in improving the cost-efficiency and performance of multi-clouds. The study examined a thousand instances of services provided by AWS, Azure, Google Cloud, and IBM, which allowed building a framework of the approximate modelling that incorporates feature engineering, predictive machine-learning, and multi-dimensional KPI assessment. The findings showed the ability of resource allocation to be successfully forecasted with the aid of standardised classification models, where the highest results were recorded with the use of the Logistic Regression with 96% accuracy and high F1-scores by all categories of efficiency. Similar outcomes were obtained with Ensemble based algorithms like random forest, XGBoost and gradient boosting, with all showing that the predictive quality was consistent and suggests that patterns of efficiency classes based on utilisation, throughput, latency and QoS metrics are learnable patterns in cloud service behaviour.

The KPI and cost efficiency analysis also indicated that the cloud providers are very different when it comes to throughput, latency, resource usage, and performance to cost ratio. AWS and Azure typically shared equal cost-performance features and Google cloud showed good network performance and IBM was competing on locality and energy measures. These results support the applicability of the designed KPIs, in particular, Resource Efficiency Score, Cost Proxy, Energy Proxy, and Locality Score, when developing a valuable foundation in decision-making in heterogeneous cloud infrastructures.

All in all, the research proves that resource allocation using machine-learning can be considered a feasible solution when it comes to studying and optimising cloud performance. The developed reproducible pipeline, combined with the CloudResourceAllocator model, presents a feasible structure that can be expanded or implemented in practice as cloud management systems. Although the simulated workload dynamics and cross-sectional data represent the limited aspects of the research, it is well-grounded in the future intelligent cloud automation developments.

9.2 Future Work

The further development of work can be carried out in the following directions which would contribute to the increase of practicality, intelligence, and adaptability of resource allocation model. The first direction is to extend the dataset with the actual workload traces, which address the time variations, peak-loads, VM migrations, and auto-scaling behaviours. This kind of longitudinal data would facilitate time series learning and allow dynamic prediction of the efficiency variation even before resource bottlenecks develop.

The second direction of development is the adaptation of deep-learning structures and the use of reinforcement learning techniques. Neural networks like LSTMs may simulate

the temporal correlation of work load sequences, and reinforcement learning agents may acquire the best policy to distribute the resources under different cost and performance constraints, particularly in the real-time auto-scaling situation.

The third development can be based on the model being an integrated part of an operational cloud orchestrator. This would enable constant tracking of KPIs, auto scaling decision making, location of multi-clouds workloads, and dynamic cost management. Introducing the system using real APIs like AWS CloudWatch, Azure Monitor, and Kubernetes Metrics Server would make the existing analytical tool an intelligent decision-support solution.

Fourth extension of the future is to augment the multi-cloud optimisation layer with the ability to schedule carbon-awarely or energy-efficiently. Sustainability of the cloud is becoming a priority and the Energy Proxy metric that is used in the present study provides a basis of predicting the low-carbon resource decisions among providers.

The last possible direction of the research would be to assess fairness, reliability, and resilience-related KPIs by considering the SLA-violation prediction, network fault tolerance and anomaly detection mechanisms. This would expand the area of the model that is focused on performance-cost optimisation to the holistic operational intelligence.

References

Al-Sharafi, M. A., Iranmanesh, M., Al-Emran, M., Alzahrani, A. I., Herzallah, F., and Jamil, N. (2023). Determinants of cloud computing integration and its impact on sustainable performance in smes: An empirical investigation using the sem-ann approach. *Heliyon*, 9(5). https://doi.org/10.1016/j.heliyon.2023.e16299.

Amin, R., Costanzo, A., Alzabin, L. R., Aqdus, A., Kamarulzaman, S. F., and Alshehri, A. H. (2025). An efficient federated learning based defense mechanism for software defined network cyber threats through machine learning models. *Scientific Reports*, 15(1). https://doi.org/10.1038/s41598-025-25345-1.

Aria, M., Cuccurullo, C., and Gnasso, A. (2021). A comparison among interpretative proposals for random forests. *Machine Learning with Applications*, 6:100094. https://www.sciencedirect.com/science/article/pii/S2666827021000475.

Arista, A. (2022). Comparison decision tree and logistic regression machine learning classification algorithms to determine covid-19. *Sinkron*, 7(1):59–65. https://doi.org/10.33395/sinkron.v7i1.11243.

Bal, P. K., Mohapatra, S. K., Das, T. K., Srinivasan, K., and Hu, Y.-C. (2022). A joint resource allocation, security with efficient task scheduling in cloud computing using hybrid machine learning techniques. *Sensors*, 22(3). https://doi.org/10.3390/s22031242.

Belgacem, A., Mahmoudi, S., and Ferrag, M. A. (2022). A machine learning model for improving virtual machine migration in cloud computing. *Research Square*. Preprint.

Carneiro, T., Da Nobrega, R. V. M., Nepomuceno, T., Bian, G.-B., De Albuquerque, V. H. C., and Filho, P. P. R. (2018). Performance analysis of google colaboratory as a tool for accelerating deep learning applications. *IEEE Access*, 6:61677–61685. https://doi.org/10.1109/ACCESS.2018.2874767.

Dakić, V., Morić, Z., Kapulica, A., and Regvart, D. (2025). Analysis of azure zero trust architecture implementation for mid-size organizations. *Journal of Cybersecurity and Privacy*, 5(1):2. https://doi.org/10.3390/jcp5010002.

Gong, Y., Huang, J., Liu, B., Xu, J., Wu, B., and Zhang, Y. (2024). Dynamic resource allocation for virtual machine migration optimization using machine learning. *arXiv*. https://doi.org/10.48550/arXiv.2403.13619.

Hamedianfar, A., Maanavilja, L., and Sutinen, H. (2025). Evaluating ensemble machine learning and feature optimization for mapping peatland vegetation using uas-derived data. *International Journal of Applied Earth Observation and Geoinformation*, 144:104901. https://doi.org/10.1016/j.jag.2025.104901.

Jayaprakash, S., Nagarajan, M. D., de Prado, R. P., Subramanian, S., and Divakarachari, P. B. (2021). A systematic review of energy management strategies for resource allocation in the cloud: Clustering, optimization and machine learning. *Energies*, 14(17):5322. https://doi.org/10.3390/en14175322.

Jiang, H., Xia, Y., Yu, C., Qu, Z., and Li, H. (2025). On the implications of artificial intelligence methods for feature engineering in reliability sector with computer knowledge graph. *Alexandria Engineering Journal*, 119:587–597. https://doi.org/10.1016/j.aej.2025.01.093.

Kapoor, S. and Narayanan, A. (2023). Leakage and the reproducibility crisis in machine-learning-based science. *Patterns*, 4(9):100804. https://doi.org/10.1016/j.patter.2023.100804.

Kaul, D. (2019). Optimizing resource allocation in multi-cloud environments with artificial intelligence: Balancing cost, performance, and security. *Journal of Information and Computer Technology Education (JICTE)*, 2019(4):1–25. https://www.researchgate.net/publication/394015247_Optimizing_Resource_Allocation_in_Multi-Cloud_Environments_with_Artificial_Optimizing_Resource_Allocation_in_Multi-Cloud_Environments_with_Artificial_Intelligence_Balancing_Cost_Performance_and_Securi.

Kodi, D. (2025). Multi-cloud finops: Ai-driven cost allocation and optimization strategies. Google Scholar Citations: 3.

Kristian, A., Goh, T. S., Ramadan, A., Erica, A., and Sihotang, S. V. (2024). Application of ai in optimizing energy and resource management: Effectiveness of deep learning models. *International Transactions on Artificial Intelligence*, 2(2):99–105.

Li, H., Wang, G., Li, L., and Wang, J. (2024). Dynamic resource allocation and energy optimization in cloud data centers using deep reinforcement learning. *Deleted Journal*, 1(1):230–258. Google Scholar Citations: 23.

Mahida, A. (2022). Comprehensive review of optimizing resource allocation in cloud computing for cost efficiency. *Journal of Artificial Intelligence & Cloud Computing*, pages 1–4. https://doi.org/10.47363/jaicc/2022(1)232.

Mungoli, N. (2023). Scalable, distributed ai frameworks: Leveraging cloud computing for enhanced deep learning performance and efficiency. Google Scholar Citations: 123.

Naga, C. (2023). Comparative analysis of public cloud providers for big data analytics: Aws, azure, and google cloud. *International Journal For Multidisciplinary Research*, 5(3). https://doi.org/10.36948/ijfmr.2023.v05i03.52071.

Nawrocki, P. and Osypanka, P. (2021). Cloud resource demand prediction using machine learning in the context of qos parameters. *Journal of Grid Computing*, 19(2). https://doi.org/10.1007/s10723-021-09561-3.

Osypanka, P. and Nawrocki, P. (2020). Resource usage cost optimization in cloud computing using machine learning. *IEEE Transactions on Cloud Computing*, 10(3):1–1. https://doi.org/10.1109/TCC.2020.3015769.

Prasad, V. K., Dansana, D., Bhavsar, M. D., Acharya, B., Gerogiannis, V. C., and Kanavos, A. (2023). Efficient resource utilization in iot and cloud computing. *Information*, 14(11):619. https://doi.org/10.3390/info14110619.

Raschka, S., Patterson, J., and Nolet, C. (2020). Machine learning in python: Main developments and technology trends in data science, machine learning, and artificial intelligence. *Information*, 11(4):1–44. https://doi.org/10.3390/info11040193.

Sabyasachi, S. and Muppala, J. (2022). Energy-efficient virtual machine placement and sla-aware resource management in cloud data centers. *Journal of Cloud Computing*, 11(1):1–15.

Trindade, S., Bittencourt, L. F., and da Fonseca, N. L. S. (2022). Resource management at the network edge for federated learning. *Digital Communications and Networks*. https://doi.org/10.1016/j.dcan.2022.10.015.

Urblik, L., Kajati, E., Papcun, P., and Zolotova, I. (2023). A modular framework for data processing at the edge: Design and implementation. *Sensors*, 23(17):7662. https://doi.org/10.3390/s23177662.

Zhou, G., Tian, W., Buyya, R., Xue, R., and Song, L. (2024). Deep reinforcement learning-based methods for resource scheduling in cloud computing: Review and future directions. *Artificial Intelligence Review*, 57(5). Google Scholar Citations: 127.