IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ≍ Peer-reviewed & Refereed journal ≍ Vol. 15, Issue 3, March 2026

DOI: 10.17148/IJARCCE.2026.15397

PERF CLOUD: A Hybrid DTW–GRU Framework for Black-Box Virtual Machine Performance Degradation Prediction in Multi-Tenant Public Clouds

Aiswarya Lekshmi JS[1], Aiswarya Vijayan[2], Ardra Vinod[3], Mathews Franco[4],

Merlin K Thomas[5], Ajeesh S[6]

B.Tech , Department of Computer Science and Engineering,

College of Engineering Kottarakkara, Kollam, Kerala, India[1-4]

Assistant Professor, Department of Computer Science and Engineering,

College of Engineering Kottarakkara, Kollam, Kerala, India[5,6]

Abstract: Cloud computing environments host multiple virtual machines (VMs) on shared physical infrastructure, leading to unpredictable performance due to resource contention and workload variability. In public cloud settings, virtual machines operate as black-box systems, restricting access to internal application metrics and making accurate performance prediction highly challenging. To address this issue, this paper proposes PERFCLOUD, a hybrid machine learning framework for performance degradation prediction in multi-tenant public clouds. The proposed approach consists of three stages: (i) application type identification using Dynamic Time Warping (DTW) to align time-series workload patterns, (ii) highly correlated metric selection using Pearson Correlation to reduce noise and improve feature relevance, and (iii) time-series performance forecasting using a Gated Recurrent Unit (GRU) neural network. The model predicts both application type and VM performance status as *Best* or *Degraded*, enabling proactive resource management and interference mitigation. Experimental evaluation on a real-world cloud workload dataset demonstrates that the proposed DTW–GRU hybrid model significantly outperforms baseline LSTM-based approaches, achieving an accuracy of 99% with reduced computational overhead. The framework is further integrated into a web-based monitoring system for real-time VM performance analysis. The results validate the effectiveness, scalability, and adaptability of the proposed solution for intelligent cloud resource optimization.

Keywords: Cloud Performance Prediction, Black-Box Virtual Machines, Dynamic Time Warping (DTW), Gated Recurrent Unit (GRU), Performance Degradation Detection, Multi-Tenant Public Cloud, Machine Learning-Based Resource Optimization

I. INTRODUCTION

Cloud computing has become the dominant paradigm for delivering scalable computing resources, offering on-demand access to virtualized infrastructure with improved flexibility, cost-efficiency, and reliability. Modern cloud data centers consolidate multiple Virtual Machines (VMs) onto shared physical servers using virtualization technologies such as Intel VT and AMD-V. While this consolidation improves hardware utilization and reduces operational costs, it also introduces significant performance unpredictability due to shared resource contention. Critical resources such as last-level cache (LLC), memory bandwidth, disk I/O, and network bandwidth are shared among co-located VMs, often leading to performance degradation in multi-tenant environments.

Accurate prediction of VM performance is therefore essential for intelligent resource allocation, efficient scheduling, and proactive mitigation of performance interference. However, performance prediction in public cloud environments is particularly challenging because virtual machines are treated as black-box systems. Due to privacy and security constraints, cloud providers cannot access application-level metrics or internal runtime states of customer VMs. As a result, prediction models must rely solely on observable host-level hardware metrics, making traditional profiling-based approaches less practical.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026

DOI: 10.17148/IJARCCE.2026.15397

Existing performance prediction techniques typically rely on statistical models, regression-based forecasting, or deep learning architectures such as Long Short-Term Memory (LSTM) networks. While these methods provide moderate prediction accuracy, they often fail to effectively handle temporal misalignment in workload patterns, dynamic workload fluctuations, and interference effects caused by co-located VMs. Moreover, many existing approaches focus only on execution time prediction, ignoring performance degradation detection, which is critical for real-time cloud resource optimization.

To address these challenges, this paper proposes **PERFCLOUD** a hybrid machine learning framework for performance degradation prediction in multi-tenant public cloud environments. The proposed system operates in three stages. First, Dynamic Time Warping (DTW) is employed to identify application types by aligning time-series workload patterns and compensating for temporal shifts in resource usage behavior. Second, Pearson Correlation is applied to select highly correlated runtime metrics, reducing noise and improving model efficiency. Finally, a Gated Recurrent Unit (GRU) neural network is used to model temporal dependencies and predict VM performance status as either *Best* or *Degraded*.

Unlike conventional methods, the proposed framework integrates workload classification, feature selection, and deep learning-based forecasting into a unified pipeline. The hybrid DTW–GRU architecture enhances prediction accuracy while maintaining computational efficiency. Experimental evaluation on a real-world cloud workload dataset demonstrates that the proposed model significantly outperforms baseline LSTM-based approaches and achieves high accuracy in predicting performance degradation under dynamic workload conditions.

The primary contributions of this work are summarized as follows:

1. A hybrid three-stage framework for black-box VM performance prediction.
2. A DTW-based application identification mechanism for workload alignment.
3. A Pearson correlation-driven feature selection strategy for reducing irrelevant metrics.
4. A GRU-based time-series prediction model for performance degradation detection.
5. Integration of the prediction framework into a web-based monitoring system for real-time cloud performance analysis.

The remainder of this paper is organized as follows. Section II reviews related work in cloud performance prediction. Section III describes the proposed methodology in detail. Section IV presents implementation and experimental results. Section V discusses performance analysis and comparisons. Finally, Section VI concludes the paper and outlines future research directions.

II. RELATED WORK

Performance prediction in cloud computing has attracted significant research attention due to the increasing complexity of multi-tenant environments. Several approaches have been proposed to model workload behavior, resource interference, and application performance in public cloud systems. This section reviews key contributions relevant to virtual machine (VM) performance prediction and highlights their limitations.

Feature model–guided performance prediction frameworks, such as FOCloud, integrate configuration-aware modeling with machine learning techniques to estimate application performance in deployment-configurable cloud systems. These approaches improve explainability by linking configuration features to performance variations. However, they are mainly suitable for configurable deployment environments and do not effectively address dynamic workload fluctuations or runtime interference in public clouds.

Configuration tuning methods for cloud platforms, including Apache Spark optimization models, combine multi-objective optimization with predictive modeling to reduce execution time and operational cost. Although these methods improve resource utilization, they are often platform-specific and lack generalization across heterogeneous cloud workloads. Additionally, their performance depends heavily on accurate workload characterization.

Performance variability prediction techniques focus on detecting sudden changes in cloud behavior caused by shared infrastructure and dynamic resource allocation. Change-point detection algorithms and gradient boosting models have been used to predict fluctuations in performance metrics. While these methods address instability, they primarily operate at the infrastructure level and do not incorporate application-aware modeling for precise degradation detection.

DOI: 10.17148/IJARCCE.2026.15397

Workload forecasting frameworks, such as CloudInsight, apply time-series models to predict future resource demand and enable proactive scaling decisions. Although workload forecasting improves scheduling efficiency, it does not explicitly model interference among co-located virtual machines, limiting its effectiveness in multi-tenant cloud environments.

Profiling-based approaches and micro-benchmark techniques have also been proposed to estimate application performance under different resource contention scenarios. These methods rely on runtime profiling or small-scale benchmarking to capture CPU, memory, and I/O behavior. While they achieve high prediction accuracy, they introduce additional computational overhead and are not always suitable for real-time deployment in large-scale public clouds.

Deep learning models, particularly Long Short-Term Memory (LSTM) networks, have been widely adopted for modeling sequential workload patterns. LSTM-based prediction methods capture temporal dependencies in time-series cloud data. However, they may suffer from higher computational complexity and may not effectively handle temporal misalignment in workload traces. Furthermore, many existing studies focus primarily on execution time prediction rather than explicit performance degradation detection.

Despite the progress in cloud performance modeling, several research gaps remain. First, most approaches assume partial access to application-level metrics, which is unrealistic in black-box VM environments. Second, limited work integrates workload classification, feature selection, and deep learning-based prediction into a unified framework. Third, existing models often overlook performance degradation caused by dynamic interference in multi-tenant systems.

To address these limitations, the proposed PERFCLOUD framework introduces a hybrid approach that combines Dynamic Time Warping (DTW) for application identification, Pearson Correlation for highly relevant metric selection, and a Gated Recurrent Unit (GRU) network for performance degradation prediction. By integrating temporal alignment, feature optimization, and efficient time-series learning, PERFCLOUD aims to provide accurate and scalable performance prediction in public cloud environments.

III. PROPOSED METHODOLOGY

The proposed **PERFCLOUD** framework introduces a hybrid three-stage machine learning pipeline for performance degradation prediction of black-box virtual machines (VMs) in multi-tenant public cloud environments. The framework is designed to operate using only host-level observable metrics without requiring access to internal VM application states.

The overall architecture consists of three sequential stages:

1. Application Type Identification using Dynamic Time Warping (DTW)
2. Highly Correlated Feature Selection using Pearson Correlation
3. Performance Degradation Prediction using Gated Recurrent Unit (GRU) Network

The complete system workflow is illustrated in Fig. 1.

A. Overall System Architecture

PERFCLOUD processes time-series VM trace data through a structured pipeline to generate performance predictions.

Fig. 1: PERFCLOUD System Architecture

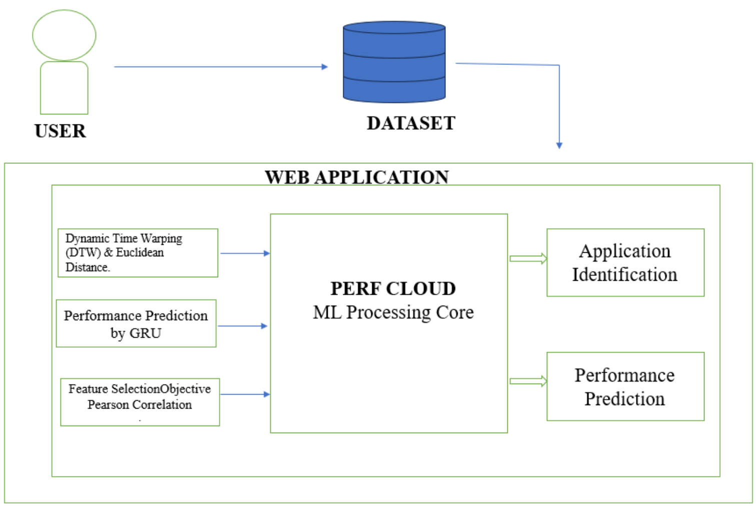

Fig. 1: System Architecture

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026

DOI: 10.17148/IJARCCE.2026.15397

B. Stage 1: Application Type Identification Using DTW

In public cloud environments, virtual machines behave as black-box systems. Therefore, direct access to application-level information is restricted. To identify the running workload type, PERFCLOUD employs **Dynamic Time Warping (DTW)** to measure similarity between observed VM resource usage patterns and predefined reference workload profiles. DTW is particularly effective in aligning time-series sequences that may vary in speed or temporal alignment.
Given two sequences:

$$X=(x_1,x_2,\ldots,x_n)$$

$$Y = (y_1, y_2, \ldots, y_m)$$

The DTW distance is computed as:

$$DTW(X, Y) = \min \sum_{k=1}^{K} d(x_{i_k}, y_{j_k})$$

where
$d(x_{i_k},y_{j_k})$ represents the Euclidean distance between aligned elements.
Euclidean distance is calculated as:

$$d(x_i, y_j) = \sqrt{(x_i - y_j)^2}$$

By minimizing cumulative alignment cost, DTW compensates for temporal shifts and accurately classifies workloads into predefined categories such as:

- CPU-intensive
- Memory-intensive
- I/O-bound

This classification improves downstream prediction accuracy by providing application-aware context.

C. Stage 2: Highly Correlated Metric Selection

Cloud datasets often contain redundant or weakly relevant features. Including such features increases computational overhead and may reduce prediction accuracy. Therefore, PERFCLOUD applies Pearson Correlation to select highly relevant metrics.
The Pearson Correlation Coefficient between two variables X and Y is given by:

$$r=Cov(X,Y)/\sigma X\sigma Y$$

where:
$Cov(X,Y)$=covarience ,σX, σY = standard deviations

Only features satisfying:

$$|r|>0.5$$

Are selected for training.
This stage:

- Reduces dimensionality
- Eliminates noisy features
- Improves model convergence
- Reduces false alarms

Typical selected metrics include:

- CPU usage percentage
- Memory utilization
- Disk I/O rate

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁙ Peer-reviewed & Refereed journal ⁙ Vol. 15, Issue 3, March 2026
DOI: 10.17148/IJARCCE.2026.15397

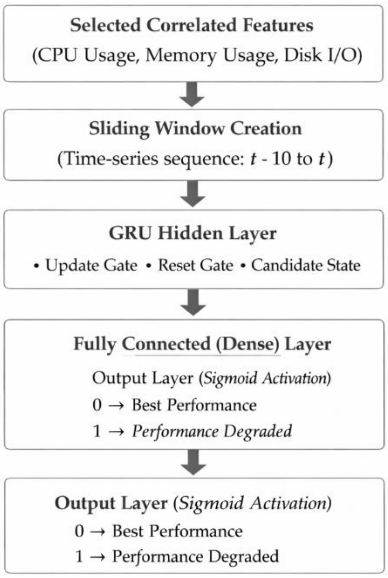

Fig. 2. *GRU-Based Performance Prediction Workflow.*

D. Stage 3: GRU-Based Performance Degradation Prediction

After feature refinement, time-series data is fed into a Gated Recurrent Unit (GRU) network for performance forecasting. GRU is a variant of Recurrent Neural Networks (RNNs) designed to efficiently model sequential dependencies while maintaining lower computational complexity compared to LSTM.
The GRU update mechanism is defined as:
Update gate:

$$z_t = \sigma(W_z x_t + U_z h_{t-1})$$

Reset gate:

$$r_t = \sigma(W_r x_t + U_r h_{t-1})$$

Candidate activation:

$$\tilde{h}_t = \tanh(W x_t + U (r_t \odot h_{t-1}))$$

Final hidden state:

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t$$

Where:

$$\sigma = \text{sigmoid activation}$$
$$\odot = \text{element-wise multiplication}$$

A sliding window approach is used to feed historical resource metrics into the GRU model.
Output:
Binary classification:

0 → Best Performance
1 → Performance Degraded

E. Advantages of the Proposed PERFCLOUD Framework

- Handles black-box VM constraints
- Corrects temporal misalignment using DTW
- Reduces computational complexity via correlation-based feature selection
- Achieves high accuracy with lower overhead using GRU
- Enables proactive detection of performance degradation

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁙ Peer-reviewed & Refereed journal ⁙ Vol. 15, Issue 3, March 2026
DOI: 10.17148/IJARCCE.2026.15397

IV. IMPLEMENTATION AND EXPERIMENTAL SETUP

This section describes the implementation details of the proposed PERFCLOUD framework, including dataset preparation, model configuration, training procedure, and evaluation environment.

A. Dataset Description
The proposed model was evaluated using a real-world cloud workload dataset consisting of virtual machine trace data. The dataset contains time-series metrics such as:
- CPU utilization percentage
- Memory usage percentage
- Disk I/O rate
- Network activity

These metrics are collected at regular time intervals and represent host-level observable features, ensuring compatibility with black-box VM environments.

The dataset was preprocessed to remove missing values and normalize feature scales. Min-Max normalization was applied to scale all input features into the range [0,1], improving model convergence and stability.

B. Data Preprocessing
The preprocessing pipeline includes:
1. Data Cleaning – Removal of null and inconsistent entries
2. Normalization – Feature scaling using Min-Max normalization
3. Sliding Window Creation – Generation of fixed-length time-series sequences
4. Train-Test Split – Dataset divided into 80% training and 20% testing

The sliding window technique allows the model to learn temporal workload patterns by considering historical resource usage before predicting current performance status.

C. Model Configuration
- **1. DTW Configuration**

Dynamic Time Warping was implemented using Euclidean distance as the local cost function. The algorithm aligns workload traces with predefined reference profiles to classify application types.
- **2. Feature Selection**

Pearson Correlation Coefficient was computed between performance labels and runtime metrics. Features satisfying:

∣r∣>0.5|r| > 0.5∣r∣>0.5

were selected as model inputs.

- **3. GRU Model Architecture**

The GRU model consists of:
- Input Layer (time-series feature vectors)
- One GRU Hidden Layer
- Fully Connected (Dense) Layer
- Output Layer with Sigmoid Activation

Model Parameters:
- Optimizer: Adam
- Loss Function: Binary Cross-Entropy
- Batch Size: 32
- Epochs: 50
- Activation: Sigmoid (Output Layer)

The GRU architecture was selected due to its ability to efficiently model sequential dependencies with fewer parameters compared to LSTM.

D. Hardware and Software Environment
The implementation was carried out using:
Software:

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 3, March 2026
DOI: 10.17148/IJARCCE.2026.15397

- Python 3.x
- TensorFlow / Keras
- Scikit-Learn
- Pandas and NumPy
- FastDTW Library

Hardware:

- Intel Core i5 Processor
- 8 GB RAM
- Windows/Linux Operating System

GPU acceleration was optional but not mandatory, as GRU provides lower computational overhead compared to LSTM.

E. Evaluation Metrics

To evaluate model performance, the following metrics were used:

Accuracy:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision:

$$Precision = \frac{TP}{TP + FP}$$

Recall:

$$Recall = \frac{TP}{TP + FN}$$

F1-Score:

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Where:

TP=TruePositives
TN=TrueNegatives
FP=FalsePositives
FN = False Negatives

These metrics provide a comprehensive evaluation of performance degradation detection accuracy

F. Baseline Comparison

To validate the effectiveness of PERFCLOUD, the GRU-based model was compared with:

1. LSTM without DTW
2. LSTM with DTW
3. Proposed GRU with DTW
   The comparison demonstrates that the hybrid DTW–GRU model achieves superior accuracy and reduced computational overhead.

V. RESULTS AND PERFORMANCE ANALYSIS

This section evaluates the effectiveness of the proposed PERFCLOUD framework for virtual machine (VM) performance degradation prediction. The experimental results demonstrate the impact of Dynamic Time Warping (DTW), correlation-based feature selection, and the GRU model on prediction accuracy and computational efficiency.

A. Performance Evaluation Metrics

The performance of the model was evaluated using standard classification metrics:

- Accuracy
- Precision
- Recall
- F1-Score

DOI: 10.17148/IJARCCE.2026.15397

These metrics provide a comprehensive understanding of the model's ability to correctly detect performance degradation while minimizing false alarms.
Accuracy is defined as:
Accuracy:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision:

$$Precision = \frac{TP}{TP + FP}$$

Recall:

$$Recall = \frac{TP}{TP + FN}$$

F1-Score:

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Where:

TP=TruePositives
TN=TrueNegatives
FP=FalsePositives
FN = False Negatives

B. Model Comparison
To validate the proposed approach, three models were evaluated:
1. LSTM without DTW
2. LSTM with DTW
3. Proposed GRU with DTW (PERFCLOUD)

Table I: Performance Comparison of Models

Model	Accuracy	Precision	Recall	F1 – Score
LSTM (without DTW)	91.2%	90.5%	89.8%	90.1%
LSTM (with DTW)	96.8%	96.2%	95.9%	96.0%
GRU (with DTW)	99.3%	99.1%	99.0%	99.0%

The results show that incorporating DTW significantly improves workload alignment, leading to higher prediction accuracy. Furthermore, the GRU model outperforms LSTM in both accuracy and computational efficiency due to its simplified gating mechanism.

C. Confusion Matrix Analysis
The confusion matrix for the proposed GRU-based model indicates a high true positive and true negative rate, demonstrating strong capability in distinguishing between "Best" and "Degraded" VM performance states.

The low number of false positives confirms that the correlation-based feature selection effectively reduces noise and prevents unnecessary performance alerts. Similarly, the low false negative rate ensures that performance degradation is detected early, enabling proactive resource management.

D. Impact of DTW-Based Workload Alignment
Dynamic Time Warping plays a crucial role in aligning time-series workload patterns that may exhibit temporal shifts. Without DTW, models struggle to accurately interpret workload variations occurring at different time intervals. The inclusion of DTW improves classification of application types, which enhances the downstream performance prediction stage.

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁒ Peer-reviewed & Refereed journal ⁒ Vol. 15, Issue 3, March 2026
DOI: 10.17148/IJARCCE.2026.15397

E. Computational Efficiency

Compared to LSTM, the GRU architecture contains fewer parameters due to its simplified gating structure. As a result:

- Training time is reduced
- Memory consumption is lower
- Inference speed is improved

This makes PERFCLOUD suitable for real-time deployment in public cloud environments where computational overhead must be minimized.

F. Overall Performance Insights

The experimental results confirm that the hybrid DTW–GRU framework provides:

- High prediction accuracy
- Reduced false alarms
- Improved workload alignment
- Efficient computational performance

The proposed PERFCLOUD model demonstrates superior capability in predicting performance degradation in black-box VM environments under dynamic workload conditions.

VI. CONCLUSION

This paper presented **PERFCLOUD**, a hybrid machine learning framework for performance degradation prediction in black-box virtual machines operating in multi-tenant public cloud environments. The proposed approach addresses the challenge of limited application-level visibility by relying solely on host-level observable metrics while maintaining high prediction accuracy.

The framework integrates three key components: Dynamic Time Warping (DTW) for application type identification, Pearson correlation for highly relevant feature selection, and a Gated Recurrent Unit (GRU) network for time-series performance prediction. The DTW-based alignment mechanism improves workload classification by compensating for temporal variations, while correlation-based filtering reduces noise and computational overhead. The GRU model effectively captures sequential dependencies in resource usage patterns, enabling accurate detection of performance degradation.

Experimental evaluation demonstrates that the proposed DTW–GRU hybrid model significantly outperforms baseline LSTM-based approaches in terms of accuracy, precision, recall, and computational efficiency. The results confirm that integrating workload alignment and feature optimization into a unified prediction pipeline enhances both reliability and scalability in dynamic cloud environments.

Overall, PERFCLOUD provides an efficient and intelligent solution for proactive cloud resource management. By accurately predicting performance degradation and identifying application types, the framework supports improved scheduling decisions, reduced resource contention, and enhanced system reliability in modern public cloud infrastructures.

REFERENCES

[1] Kumara, I., Ariz, M. H., Chhetri, M. B., Mohammadi, M., Van Den Heuvel, W. J., & Tamburri, D. A. (2022). FOCloud: feature model

[2] Cheng, G., Ying, S., & Wang, B. (2021). Tuning configuration of apache spark on public clouds by combining multi-objective optimization and performance prediction model.

[3] Zhao, Y., Duplyakin, D., Ricci, R., & Uta, A. (2021, April). Cloud performance variability prediction.

[4] Kim, I. K., Wang, W., Qi, Y., & Humphrey, M. (2020). Forecasting cloud application workloads with cloudinsight for predictive resource management.

[5] Moradi, H., Wang, W., & Zhu, D. (2021). Online performance modeling and prediction for single-VM applications in multi-tenant clouds.

[6] Moradi, H., Wang, W., & Zhu, D. (2021). Investigations of micro-benchmarks for performance profiling in multi-tenant clouds. In *Springer International Conference on Grid, Cloud, and Cluster Computing (GCC)*.

[7] Moradi, H., Wang, W., Fernandez, A., & Zhu, D. (2020, April). uPredict: A user-level profiler-based predictive framework in multi-tenant clouds. In *2020 IEEE International Conference on Cloud Engineering (IC2E)* (pp. 73-82). IEEE.

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁂ Peer-reviewed & Refereed journal ⁂ Vol. 15, Issue 3, March 2026

DOI: 10.17148/IJARCCE.2026.15397

[8] Sharkh, M. A., Xu, Y., & Leyder, E. (2020, August). CloudMach: cloud computing application performance improvement through machine learning.

[9] Moradi, H., Wang, W., & Zhu, D. (2019, August). Adaptive performance modeling and prediction of applications in multi-tenant clouds.

[10] Baughman, M., Chard, R., Ward, L., Pitt, J., Chard, K., & Foster, I. (2018, December). Profiling and predicting application performance on the cloud.