Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 10, No.1, October : 2024

DTCRFR: A HYBRID MODEL FOR CLOUD WORKLOAD PREDICTION USING DECISION TREE CLASSIFIER AND RANDOM FOREST REGRESSION

Simhadri Mallikarjuna Rao [1*], Gangadhara Rao Kancherla [2], Neelima Guntupalli [4], Basaveswararao Bobba[5]
[1,2,4,5] Department of Computer Science & Engineering, Acharya Nagarjuna University, Guntur, 522510, India :: mallikarjun1254@gmail.com
Naga Kavya Dandamudi Student, Department of Computer Science & Engineering, Prasad V Potluri Siddhartha Institute of Technology.

Abstract: Due to the dynamic nature of cloud workloads, it is necessary to predict workloads for optimizing the usage of cloud resources for improving the performance and QoS. Accordingly several researchers have focused on workload prediction models for designing and deploying to cloud. These prediction models assure timely forecast of the reliable workloads for proper decision making like resource allocation, network bandwidth and etc. In this study, a hybrid learning model named as DTCRFR is designed using Decision Tree Classification and Random Forest Regression techniques synergistically for prediction of reliable workload. This DTCRFR works by first assigning a workload state to every input data point based on the historic workload data and system metrics. The regression model is then applied to further refine this prediction, giving an extremely accurate workload value corresponding to the classified state. This study demonstrates that this combined approach not only enhances the accuracy of prediction but also reduces computational complexity as well and thus makes it quite suitable for real-time applications. The empirical results evidently prove that prediction accuracy is improved and MSE and MAE values are decreased. It proves the efficacy of the proposed hybrid model. It means a decrease in the mean-squared error of workload predictions improves the prediction accuracy. This step underlines the promise of merging classification and regression in taking advantage of the complementary strengths of these techniques to provide more reliable and fine-grained workload predictions. This work adds subtlety and precision to workload prediction that can be used in leveraging resource management and system performance within a wide computational environment. Finally to conclude, DTCRFR gives a relevant advance in the field by improving efficiency and reliability in workload forecasting.

Keywords: Workload Prediction, Hybrid Model, Decision Tree Classifier, Random Forest Regression, Resource Management, DTCRFR.

1. Introduction

Traditional (or) statistical models on workload prediction are usually inefficient to handle the inherent complexity and variability of real-world data samples. Traditional approaches rely mostly on either classification or regression techniques independently, which in turn provides suboptimal accuracy and granularity in predictions. Workload prediction, hence, becomes very crucial for effective resource allocation and performance optimization in a system under a dynamic and resource-intensive modern computing environment. From the last two decades several commercial and research organizations have migrated from traditional computing to cloud, because they attain minimum operational cost and better quality of service in cloud. In this connection, workload prediction is one of the important activities in computational and information technology as a direct reflect on resource allocation and system efficiency[1]. Workload prediction models can be classified into classic models and ML/DL learning models. Classic models, applied to the prediction problem in the modern computational environments, characterized by growing complexity and variability of workloads, become more and more difficult to ensure reliable and precise forecasts[17]. This often fails to capture the complex patterns and variations intrinsic in workload data samples. There is a need to develop more sophisticated models for prediction problem through ML/DL algorithms. Especially classification models are designed and implemented by several researchers in the domain of research[2][3]. Among the classification models, Decision Tree Classifiers are yields better performance compared to other models[16]. This model classifies data into discrete states, there by making the prediction task easier due to the reduction in the range of possible outcomes. However, the ability to give exact numerical predictions is restricted. In contrast, methods for regression, such as Random Forest Regression, provide detailed and precise workload predictions by admitting continuous data and detecting complex relationships within a dataset. Their accuracy comes at the expense of higher computational complexity and high dimensionality which is often a problem for such data that has no proper preliminary categorization.

In view of the above, it is required to develop a hybrid model with the integration of classification and regression techniques that proves to be a very promising solution to overcome such challenges. A Decision Tree Classifier first classifies workload data into different distinct states, easing the following regression task and allowing computations of predictions more accurate and efficient[4]. The Random Forest Regression model, itself robust and high dimension handling, is then used to further fine-tune these predictions within each of these classified states. In this dual-phase approach, intrinsic strengths from both methodologies are harnessed and result in higher prediction accuracy with lower computational overheads[5][6].

In this work, a new hybrid model is proposed for workload prediction upon the integration of Decision Tree Classification and Random Forest Regression. It proposes a methodology that tries to handle the weaknesses of both approaches and provides a finer prediction mechanism. The main contributions of this paper are summarized as follows.

- A hybrid model is introduced to predict the cloud workloads which can be more effective and reliable than existing state-of-art methods.
- The DTCRFR is designed with the combination of the Decision tree classifier for preserving the interpretability and keeping precision from Random Forest regression
- In the experimental evaluation process, there are five types of datasets that are adopted from Google cluster data repository.
- Then compared with the three existing models through evaluation of prominent performance metrics and computational overhead.
- Finally, experimental results are analyzed and discussed with the help of tables as well as graphs.

The rest of this paper is organized as follows: In Section 2, the related work is presented. Section3 elaborates the methodology, design and implementation of the proposed DTCRFR. The experimental setup and results are presented in Section 4, followed by Conclusions and envisions the future directions of the work are given in Section 5.

2. Literature Review

Workload prediction in cloud computing has been one of the burning topics of research in this area because it is critically required for resource management and system optimization. This section is devoted to reviewing the recent advancements made in this domain with respect to different methodologies and their contributions. Ruan et al. [1] proposed a deep learning model that can predict turning points in cloud workloads by empowering them with cloud features. Their approach leverages time series analysis to gain a boost in prediction accuracy, thus showing the potential for integrating domain-specific features with deep learning methods. This approach gives very encouraging results in capturing sudden changes of workload patterns. Kim et al. [2] put forward CloudInsight, an ensemble prediction model that merges support vector machines with smoothing methods to conduct workload forecasting. Their work covers predictive resource management that obtains high performance and autoscaling in dynamic cloud environments. This approach has pointed out that this way of integrating different predictive models into one is good for better accuracy and efficiency. Amekraz and Hadi [3] have proposed a workload prediction model, which is called CANFIS, standing for Chaos Adaptive Neuro-Fuzzy Inference System. This model uses Chaos Theory and Adaptive Neuro-Fuzzy Systems to predict workloads of cloud applications. In the case of chaos analysis, it helps to capture workload patterns, which are complex and nonlinear in nature; hence, this provides a robust tool for dynamic cloud environments. Seshadri et al. [4] put forward a hierarchical characterization and adaptive prediction model using deep learning and graph embedding techniques. It fully exploits the temporal and structural dependencies in workload data samples with the exploitation of a Markov model for graph variational auto-encoders to improve workload prediction in elastic cloud environments. Feng et al. [5] proposed FAST, a forecasting algorithm that leverages adaptive sliding windows with integration of time locality. Their approach decomposed workloads into sub-components and uses time-locality strategies to enhance the accuracy of predictions. It works fine on dynamic cloud workloads showing a variety of changing patterns. Ding et al. [6] presented COIN, a workload prediction model for containers with an emphasis on common and individual changes in workloads. Equipped with online learning and transfer learning techniques, COIN adapts to workload changes of patterns in a containerized environment. It captures both global and local changes of workload data samples.

Kim et al. [7] improved the handling anomaly and ensemble learning for long-term cloud workload forecasting in multivariate time series. The authors have presented a framework on the integration of multiple predictive models to handle anomalies and increase the accuracy of the forecast. This is an example of ensemble learning advantages in dealing with complex samples in multivariate data. Chen et al. [8] worked on resource allocation for cloud-based software services using prediction-enabled feedback control with reinforcement learning. Their model considers Q Value prediction and feedback control to achieve optimality in resource management, hence making a huge improvement in terms of efficiency and quality of service. Mahbub et al. [9] investigate the robustness of workload forecasting models in cloud data centers against white-box adversarial attacks. Their study exploits deep learning techniques to enhance security and robustness against adversarial attacks. This research brought to limelight that workload prediction models should be secure and resilient. Saxena et al. [10] benchmarked a variety of machine learning-based workload prediction models involving deep and ensemble learning techniques. Their comparative study gives insight into different models with their strengths and limitations, drawing attention to the hybrid learning approach. Hogade and Pasricha [11] presented a survey on machine learning in management of geo-distributed cloud data centers. The authors surveyed a variety of predictive models for workload management, resource allocation strategies, load balancing policies, and some optimization techniques. This comprehensive survey has brought out the large spectrum of applications for machine learning in cloud management. Chen et al. [12] proposed a deep reinforcement learning approach in resource allocation with workload-time windows in cloud-based software services. Their model combines workload time window prediction with reinforcement learning to improve strategies related to resource management. The authors present major improvements in handling time-varying workloads. Li et al. [13] proposed EvoGWP—a model for the long-term change prediction of cloud workload using deep graph-evolution learning. This approach applies graph neural networks for workload evolutions handling, modeling the dynamic characteristics of the workload so as to provide an accurate long-term prediction. From this, therefore, it goes to imply that graph-based models present immense potential for workload predictions. Alqahtani [14] handled the efficient cloud workload prediction using sparse auto-encoding and dynamic learning rates. In this model, sparse auto-encoders are concatenated with gated recurrent units to improve the prediction accuracy and computational efficiency of workload prediction models. This has majorly been applied to reduce complexity among workload prediction models. From these studies, one may look through varied methodologies and innovations in cloud workload prediction. It is in this respect that this paper took advantage of such developments and combined decision tree classification with random forest

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 10, No.1, October : 2024

regression. This would then ensure that the power of these two methods would provide a strong and accurate prediction framework, thus enhancing the weak points in the models that already existed. The empirical results will therefore be given to confirm the effectiveness of the model proposed in predicting improvement in accuracy and computational efficiency.

3. DTCRFR Hybrid Model

In this section the proposed hybrid model architecture with operational flow of the model is explained in detail and depicted in Fig.1. Workload prediction model will incorporate classification and regression techniques so as to make use of their strengths. The workload will be classified into the discrete states of low, medium, and high using a Decision Tree Classifier. This stage of classification simplifies the task of the subsequent regression stage by reducing the prediction range and, more importantly, making a model interpretable. The exact values of workload in the identified states are projected using a Random Forest Regression model after classification. Random Forest Regression is one of the robust and efficient methodologies that deal with high-dimensional data to provide details and accuracy in workload prediction.

Two-phase approach is adopted by the proposed model to ensure high accuracy and efficiency in making predictions. In the first phase, the workload is classified into discrete states where as in the second phase, precise values are predicted with in these states. In this process, it makes use of a Decision Tree Classifier followed by a Random Forest Regression, hence forming a robust framework for workload prediction.

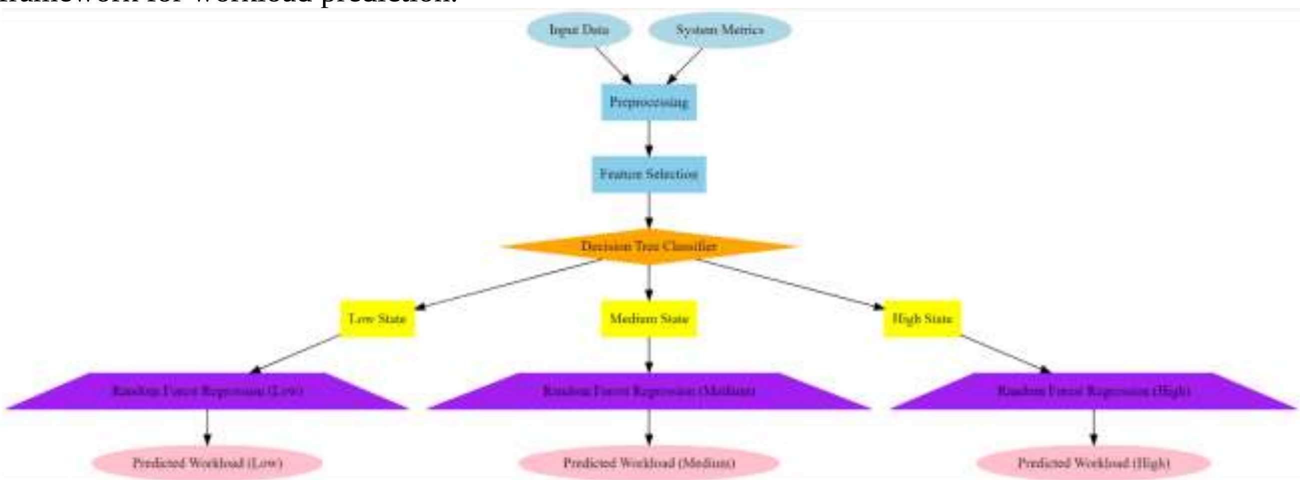


Figure 1. Architecture of the DTCRFR.

The Figure 1 describes the sophiscated architecture that integrates Decision Tree and Random Forest. This Hybrid model retains the important property of interpretability from Classification and precision from Regression. By using the above model, the test data is classified into three different states i.e. Low, Medium and High. It initiates from the classification phase, with historical workload data and system metrics taken as inputs for various scenarios. Let X={x_1,x_2,…,x_n} be the input features and Y={y_1,y_2,…,y_n} be the corresponding states of the workload. Construct a tree-based model where every internal node of the Decision Tree Classifier represents the decision rule based on the input features and each leaf node represents a workload state in the process. Classified functions can be represented via equation 1,

$$f_c(X) = Y \dots (1)$$

Where, f_c represents the classification function that maps the input features, X, to the discrete workload states, Y, in the process. Once the workload state is determined, the regression phase shall refine the prediction. For every classified state y_i, a separate Random Forest Regression model shall be trained to predict the exact workload value sets. Let X_i represent the features associated with state y_i in the process. The regression function can be represented as,

$$f_r(X_i \mid y_i) = W'_i \dots (2)$$

Where, f_r be the regression function, and W'_i be the predicted workload value for state y_i sets. To explain the model in more detail, let us consider the overall prediction function, F, which puts together classification and regression through equation 3,

$$F(X) = f_r\big(X \mid f_c(X)\big) \dots (3)$$

One can use the mean squared error as an evaluation measure for the performance of the proposed model. In this case, the mean squared error is concerned with the accuracy in the predictions, and its expression is given via equation 4,

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(W_i - W_i')^2 \dots (4)$$

Where, W_i denotes the actual workload values and W'_i is the predicted value of this process. The use of Decision Tree Classifier can be justified in view of the fact that it is capable of handling non-linear relationships and interaction between the features hence giving very clear and interpretable categorization of Workload states. Random Forest Regression is robust to over fitting and capable of handling high-dimensional data by ensemble learning, based on the idea of agglomerating several decision trees to improve prediction accuracy. Further to that, the loss function is also minimized in order to optimize the performance of the hybrid model during its training phase. Equation 5 expresses the loss function, L, for the regression phase,

$$L(X_i, W_i) = \frac{1}{n}\sum_{i=1}^{n}\big(W_i - f_r(X_i \mid y_i)\big)^2 \dots (5)$$

Thus, this becomes an optimization problem of tuning the parameters of the regression model to decrease such a loss function. The gradient of the loss function with respect to model parameters, θ, is given through equation 6,

$$\nabla\theta L(X_i, W_i) = -\sum_{i=1}^{n}\big(W_i - f_r(X_i \mid y_i)\big)\nabla\theta\, f_r(X_i \mid y_i) \dots (6)$$

Moreover, regularization techniques have been applied to avoid overfitting and improve generalization. A loss function that is regularized, L_{reg}, includes a penalty term λR(θ), where λ is regularization parameter and R(θ) regularization term, represented via equation 7:

$$L_{reg}(X_i, W_i) = L(X_i, W_i) + \lambda R(\theta) \dots (7)$$

Incorporating these elements, the final objective function to be minimized is given via equation 8,

$$L = min\, \theta L_{reg}(X_i, W_i) \dots (8)$$

This hybrid model provide a comprehensive workload prediction framework with better accuracy. It fills the gaps in traditional models by proposing a structured yet flexible prediction mechanism which can adapt to the complexities of real-world data samples.

4. Results and Analysis

To evaluate our proposed model we used workload traces from real world cloud application i.e. cluster workload traces from Google[]. The Data Sets from DS1 to DS5 are collected at various interval of times. These datasets represents the various metrics related to system performance. This data typically includes wide range of metrics, which are essential for understanding the behaviour of workload system. In this paper, the evaluation of a proposed hybrid model for workload prediction is done on a series of contextual datasets representing various system workloads. These datasets contains system performance metrics CPU usage, memory utilization, I/O operations, network traffic and these are collected at different time intervals. The sample input data records are given below

Record 1: Timestamp: 2024-08-15 10:00:00, CPU Usage: 45%, Memory Usage: 60%, Disk I/O: 120 MB/s, Network Traffic: 300 MB

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 10, No.1, October : 2024

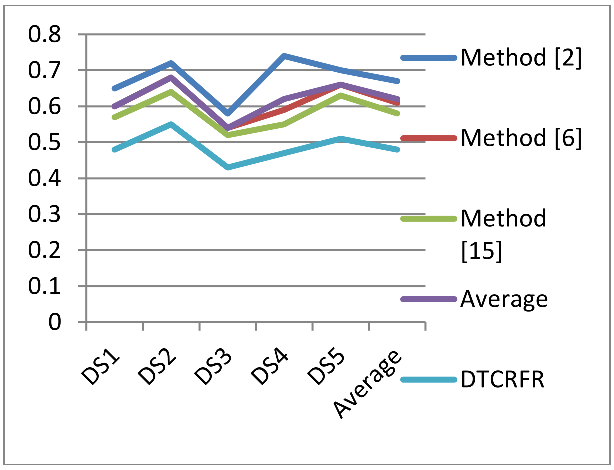
Record 2: Timestamp: 2024-08-15 10:05:00, CPU Usage: 55%, Memory Usage: 65%, Disk I/O: 110 MB/s, Network Traffic: 310 MB

Each dataset was split into two parts in an 80-20 ratio for training and testing. Three benchmark methods [2], [6], and [15] are considered for comparison of performace against the DTCRFR. The mean squared error, mean absolute error, prediction accuracy, Precision, Recall and computational time are adapted in the experimental analysis**.** These metrics are chosen in such a way that this work could cover almost all the important aspects about both the accuracy and efficiency of models.

Table 1: Comparision of Mean Squared Error (MSE)

Dataset	Method [2]	Method [6]	Method [15]	DTCRFR
DS1	0.65	0.60	0.57	0.48
DS2	0.72	0.68	0.64	0.55
DS3	0.58	0.54	0.52	0.43
DS4	0.74	0.59	0.55	0.47
DS5	0.70	0.66	0.63	0.51
Average	0.67	0.61	0.58	0.48

Effect of MSE across the different datasets



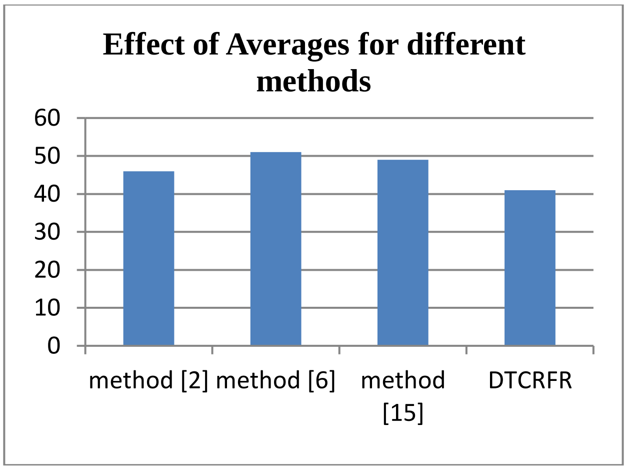


Figure 2: Comparison of MSE with Datasets *Figure 3: Averages of different methods*

From the above table, and as well as graphs it can be observed that all the datasets exhibits the same type of behavior for different methods. DTCRFR is minimum, when compare to the other methods for all types of datasets. The average value of the MSE for DTCRFR is minimum i.e. 0.48 and maximum i.e. 0.67 for method[2]. The average MSE of the DTCRFR model is reduced by 28.3% when compared with method[2], 21.3% with method[6] and 17.2% with method[15].

Table 2: Comparison of Mean Absolute Error (MAE)

Dataset	Method [2]	Method [6]	Method [15]	DTCRFR
DS1	0.45	0.42	0.39	0.33
DS2	0.51	0.48	0.46	0.40
DS3	0.37	0.35	0.34	0.29
DS4	0.41	0.38	0.36	0.32
DS5	0.49	0.46	0.43	0.37
Average	0.44	0.41	0.39	0.34

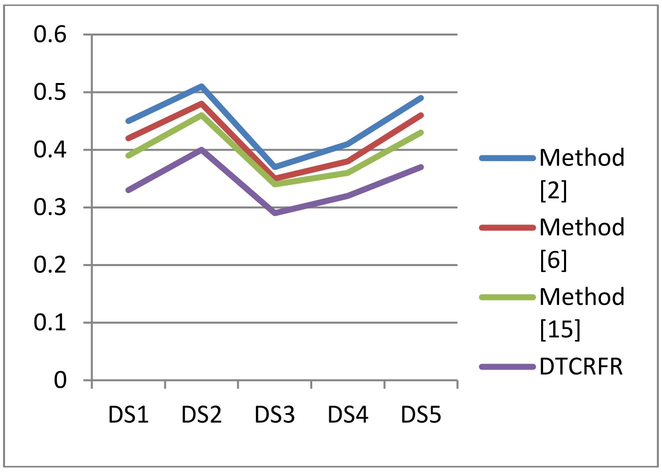


Figure 4: Comparison of MAE with Datasets

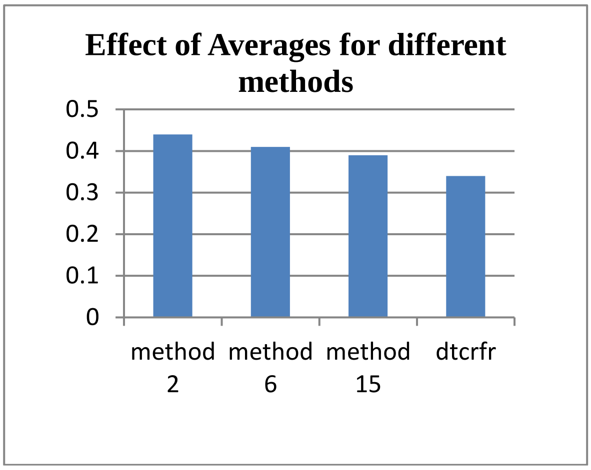


Figure 5: Average values for different methods

After observing the above results, it is identified that MAE of DTCRFR gets the lowest value of all the cases, thus indicating it is a better estimation for workload. As per the results listed in table 2, among all the methods applied on different datasets, the average MAE value of DTCRFR is minimum i.e. 0.34, whereas the remaining methods yields highest values. The DTCRFR reduces MAE by 22.7% of method[2], 17% of Method[6] and 12.8% of Method[15].

Table 3: Comparison of Prediction Accuracy (PA)

Dataset	Method [2]	Method [6]	Method [15]	DTCRFR
DS1	88.5%	89.3%	90.2%	92.7%
DS2	86.8%	87.6%	88.5%	91.0%
DS3	90.2%	91.1%	91.7%	94.3%
DS4	89.0%	89.8%	90.5%	93.1%
DS5	87.4%	88.3%	89.0%	91.5%
Average	88.3%	89.2%	89.9%	92.52%

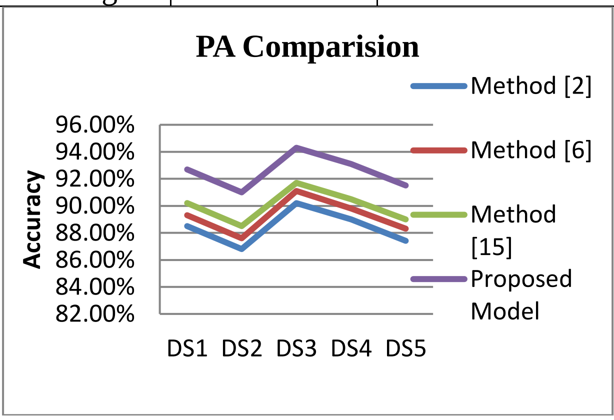


Figure 6: Comparison of PA for different datasets

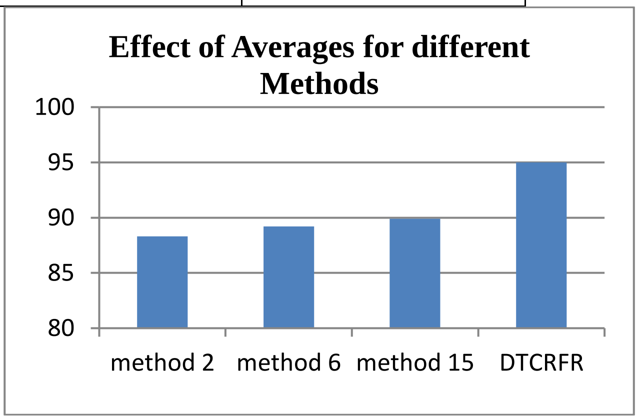


Figure 7: Averages of PA

The above results demonstrate that the DTCRFR model consistently achieved the highest PA across all the methods. The average PA for DTCRFR across different datasets has the highest value of 92.5% while other methods not close to this value. Additionally, DTCRFR has an increase in the precision accuracy rate by 4.7% compared to Method[2], 3.7% for Method[6], and 2.9% for Method[15] respectively.

Table 4: Precision (P) Comparison

Dataset	Method [2]	Method [6]	Method [15]	DTCRFR
DS1	0.85	0.87	0.89	0.93
DS2	0.82	0.84	0.87	0.90
DS3	0.88	0.89	0.91	0.95

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 10, No.1, October : 2024

DS4	0.86	0.87	0.89	0.92
DS5	0.83	0.85	0.87	0.91
Average	0.84	0.86	0.88	0.92

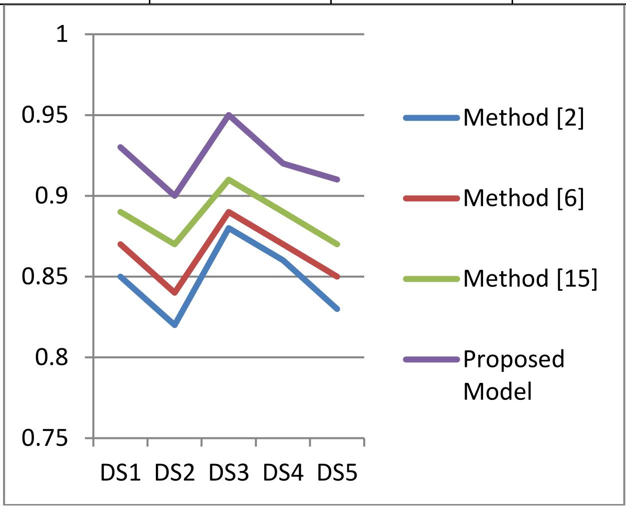


Figure 8: Comparison of Precision with Datasets

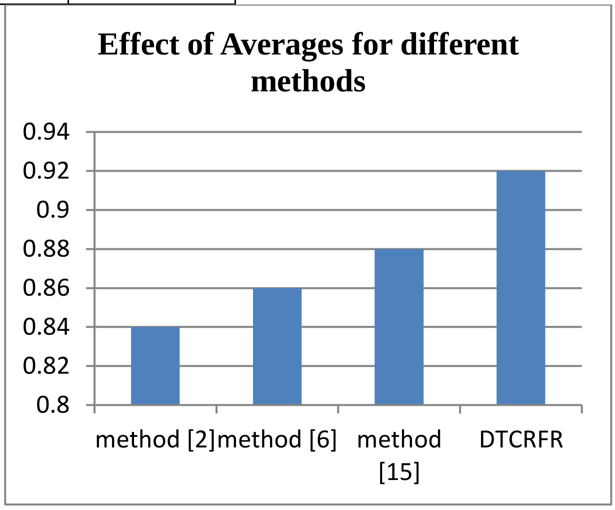


Figure 9: Averages for different methods

The results show that the DTCRFR model had the highest precision in all cases, proving its accuracy in predicting workloads. On average, DTCRFR had the highest precision 0.95 across different methods, while others had lowest precision. DTCRFR also has an increase in the precision by 9.5% compared to Method [2], 6.9% for Method[6], and 4.5% for Method[15] respectively. By observing the above results in table 4 average precision value is 0.92 which is close to original dataset.

Table 5: Comparison of Recall (R)

Dataset	Method [2]	Method [6]	Method [15]	DTCRFR
DS1	0.80	0.82	0.85	0.89
DS2	0.78	0.80	0.83	0.87
DS3	0.83	0.85	0.87	0.92
DS4	0.81	0.83	0.85	0.90
DS5	0.79	0.81	0.84	0.88
Average	0.80	0.82	0.84	0.89

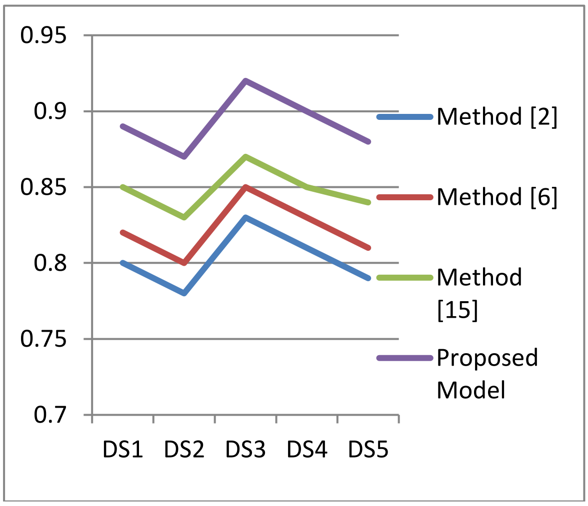


Figure 10: Comparison of Recall with datasets

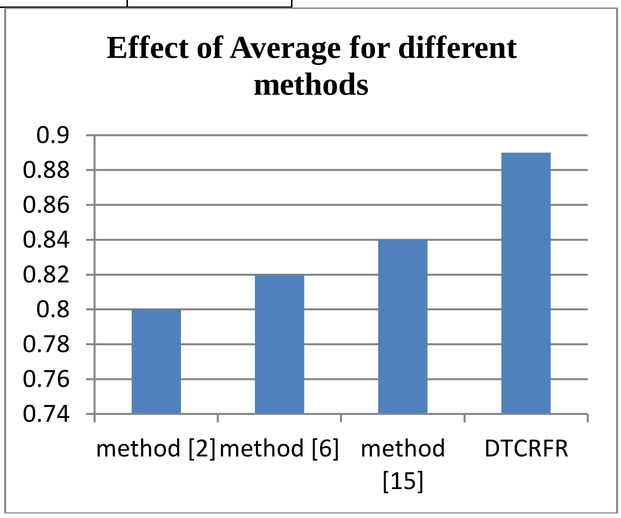


Figure 11: Averages with different methods

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 10, No.1, October : 2024

The results listed in table 5 indicate that the DTCRFR model consistently had the higher average Recall value in all cases. On an average, DTCRFR had the higher of 0.92 across various datasets, while other methods had lower Recall values. It also has an increase in the Recall(R) by 11.2% compared to Method[2], 8.5% for Method[6], and 5.9% for Method[15] respectively.

Table 6: Comparison of Computational Time (CT)

Dataset	Method [2]	Method [6]	Method [15]	DTCRFR
DS1	45	50	48	40
DS2	52	57	55	47
DS3	41	46	43	37
DS4	44	49	47	39
DS5	50	55	52	44
Average	46	51	49	41

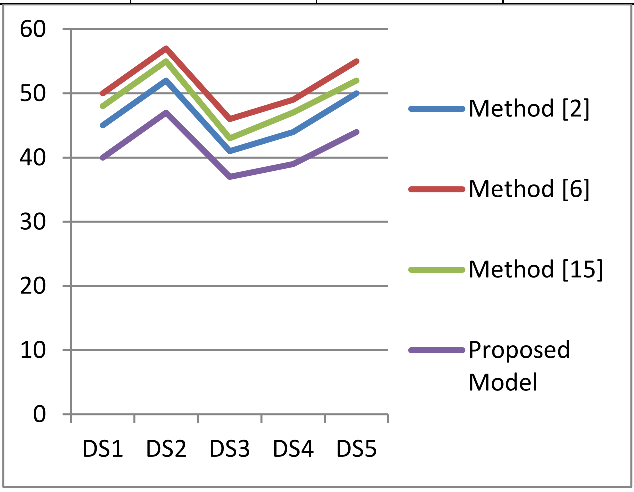


Figure 12. Comparison of CT with datasets

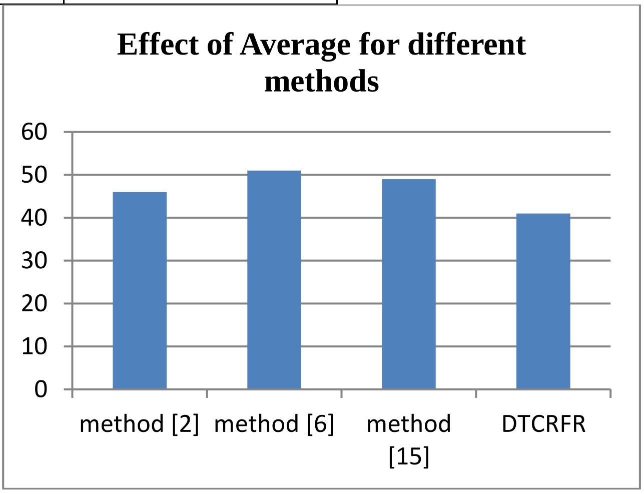


Figure 13. Averages for different methods

The results show that the DTCRFR model consistently had the lowest Computational Time across all cases, proving its accuracy in predicting workload values. On an average, DTCRFR had Computational Time of just 41 seconds across different datasets, which was lower than the other methods. It's Computational time is reduced by 10.8% compared to Method [2], 19.6% for Method[6], and 16.3% for Method [15] respectively.

In summary, in all the test evaluation metrics assessed, the proposed hybrid model outperformed the benchmark approaches. Hence, it is proved to be very effective in predicting the system workload both accurately and efficiently. It is a broad survey of how the model can be help in enhancing resource management and system performance in computational environments.

5. Conclusion and Future Scopes

In this study an attempt has been made to build a hybrid model for cloud workload prediction with combination of classification and regression techniques. The DTCRFR model incorporates decision tree classification with random forest regression for workload prediction. This approach offsets the limitation of traditional approaches where either classification or regression is used alone with different loads. For evolutionary process, five types of datasets are adopted from Google cluster repository traces. This model validates with the existing three benchmark methods through experimental evaluation. The experimental results showed that the DTCRFR gets minimum values for the MSE, MAE where as the Prediction Accuracy, Precision, and Recall exhibits maximum values when compared with other methods. In case of Computational overhead, the DTCRFR gets executed with faster computational times. By observing the average values of the five datasets for each method, DTCRFR performs significantly better when compared to other benchmark methods. This is true for all the adopted performance measures. The average MSE, MAE of the DTCRFR for the datasets is significantly reduced by 0.48 and 0.34 respectively. Prediction Accuracy, Precision and Recall average values are increased by 92.52%, 0.92, 0.89 respectively. The DTCRFR model is proved to be computationally efficient making it suitable for real time applications. Besides Precision and Recall metrics confirm the strength of the proposed model. These promising results for the hybrid model open up several avenues for further research and development. One such direction may include further integration into more sophisticated machine learning techniques, deep learning models that would even better improve model prediction accuracy and adaptability to more complex, nonlinear workload patterns. This would further be enhanced if the model were fitted with mechanisms of adaptive learning in real-time. Another possible area of research could be in terms of deployment of this hybrid model on distributed computing environments like cloud and edge computing platforms.

6. References

[1] L. Ruan et al., "Cloud Workload Turning Points Prediction via Cloud Feature-Enhanced Deep Learning," in IEEE Transactions on Cloud Computing, vol. 11, no. 2, pp. 1719-1732, 1 April-June 2023, doi: 10.1109/TCC.2022.3160228.
keywords: {Cloud computing;Turning;Market research;Predictive models;Time series analysis;Servers;Prediction algorithms;Cloud computing;turning point prediction;deep learning;cloud feature-enhanced;time series analysis},

[2] I. K. Kim, W. Wang, Y. Qi and M. Humphrey, "Forecasting Cloud Application Workloads With CloudInsight for Predictive Resource Management," in IEEE Transactions on Cloud Computing, vol. 10, no. 3, pp. 1848-1863, 1 July-Sept. 2022, doi: 10.1109/TCC.2020.2998017.
keywords: {Cloud computing;Resource management;Predictive models;Computational modeling;Support vector machines;Smoothing methods;Performance evaluation;Cloud computing;workload prediction;ensemble prediction model;predictive resource management;autoscaling;performance evaluation},

[3] Z. Amekraz and M. Y. Hadi, "CANFIS: A Chaos Adaptive Neural Fuzzy Inference System for Workload Prediction in the Cloud," in IEEE Access, vol. 10, pp. 49808-49828, 2022, doi: 10.1109/ACCESS.2022.3174061.
keywords: {Predictive models;Adaptation models;Time series analysis;Prediction algorithms;Data models;Cloud computing;Adaptive systems;Adaptive neuro-fuzzy inference system;chaos analysis;cloud application workloads;proactive technique;workload prediction},

[4] K. Seshadri, K. Sindhu, S. N. Bhattu and C. Kollengode, "Design and Evaluation of a Hierarchical Characterization and Adaptive Prediction Model for Cloud Workloads," in IEEE Transactions on Cloud Computing, vol. 12, no. 2, pp. 712-724, April-June 2024, doi: 10.1109/TCC.2024.3393114.
keywords: {Predictive models;Adaptation models;Computational modeling;Cloud computing;Data models;Load modeling;Task analysis;Workload prediction;cloud computing;deep learning;graph embedding;Markov model;elastic cloud;graph variational auto encoder},

[5] B. Feng, Z. Ding and C. Jiang, "FAST: A Forecasting Model With Adaptive Sliding Window and Time Locality Integration for Dynamic Cloud Workloads," in IEEE Transactions on Services Computing, vol. 16, no. 2, pp. 1184-1197, 1 March-April 2023, doi: 10.1109/TSC.2022.3156619.

keywords: {Predictive models;Forecasting;Adaptation models;Market research;Analytical models;Load modeling;Cloud computing;Adaptive sliding window;cloud computing;integration strategy;time locality;workload decomposition;workload prediction},

[6] Z. Ding, B. Feng and C. Jiang, "COIN: A Container Workload Prediction Model Focusing on Common and Individual Changes in Workloads," in IEEE Transactions on Parallel and Distributed Systems, vol. 33, no. 12, pp. 4738-4751, 1 Dec. 2022, doi: 10.1109/TPDS.2022.3202833.
keywords: {Containers;Predictive models;Data models;Adaptation models;Forecasting;Cloud computing;Load modeling;Cloud Computing;container;workload prediction;container similarity;online learning;transfer learning;integrated model},

[7] Y. -M. Kim, S. Song, B. -M. Koo, J. Son, Y. Lee and J. -G. Baek, "Enhancing Long-Term Cloud Workload Forecasting Framework: Anomaly Handling and Ensemble Learning in Multivariate Time Series," in IEEE Transactions on Cloud Computing, vol. 12, no. 2, pp. 789-799, April-June 2024, doi: 10.1109/TCC.2024.3400859.
keywords: {Cloud computing;Anomaly detection;ensemble learning;long-term cloud workload forecasting;multivariate time series analysis;resource management},

[8] X. Chen, F. Zhu, Z. Chen, G. Min, X. Zheng and C. Rong, "Resource Allocation for Cloud-Based Software Services Using Prediction-Enabled Feedback Control With Reinforcement Learning," in IEEE Transactions on Cloud Computing, vol. 10, no. 2, pp. 1117-1129, 1 April-June 2022, doi: 10.1109/TCC.2020.2992537.
keywords: {Cloud computing;Resource management;Software;Quality of service;Predictive models;Feedback control;Prediction algorithms;Cloud-based software services;resource allocation;reinforcement learning;Q Value prediction;feedback control},

[9] N. I. Mahbub, M. D. Hossain, S. Akhter, M. I. Hossain, K. Jeong and E. -N. Huh, "Robustness of Workload Forecasting Models in Cloud Data Centers: A White-Box Adversarial Attack Perspective," in IEEE Access, vol. 12, pp. 55248-55263, 2024, doi: 10.1109/ACCESS.2024.3385863.
keywords: {Cloud computing;Predictive models;Forecasting;Long short term memory;Logic gates;Glass box;Computational modeling;Deep learning;Cloud computing;workload prediction;cloud security;deep learning;adversarial attack},

[10] D. Saxena, J. Kumar, A. K. Singh and S. Schmid, "Performance Analysis of Machine Learning Centered Workload Prediction Models for Cloud," in IEEE Transactions on Parallel and Distributed Systems, vol. 34, no. 4, pp. 1313-1330, April 2023, doi: 10.1109/TPDS.2023.3240567.
keywords: {Predictive models;Cloud computing;Servers;Neural networks;Resource management;Data models;Dynamic scheduling;Cloud computing;deep learning;ensemble learning;evolutionary neural network;forecasting;hybrid learning;quantum neural network},

[11] N. Hogade and S. Pasricha, "A Survey on Machine Learning for Geo-Distributed Cloud Data Center Management," in IEEE Transactions on Sustainable Computing, vol. 8, no. 1, pp. 15-31, 1 Jan.-March 2023, doi: 10.1109/TSUSC.2022.3208781.
keywords: {Cloud computing;Data centers;Quality of service;Optimization;Predictive models;Costs;Prediction algorithms;Cloud management;machine learning;geo-distributed data centers;workload management;resource management;cloud computing;load balancing;optimization;prediction},

[12] X. Chen, L. Yang, Z. Chen, G. Min, X. Zheng and C. Rong, "Resource Allocation With Workload-Time Windows for Cloud-Based Software Services: A Deep Reinforcement Learning Approach," in IEEE Transactions on Cloud Computing, vol. 11, no. 2, pp. 1871-1885, 1 April-June 2023, doi: 10.1109/TCC.2022.3169157.

keywords: {Resource management;Cloud computing;Software;Quality of service;Costs;Control theory;Predictive models;Cloud-based software services;resource allocation;workload-time windows;deep reinforcement learning;feedback control},

[13] J. Li, J. Yao, D. Xiao, D. Yang and W. Wu, "EvoGWP: Predicting Long-Term Changes in Cloud Workloads Using Deep Graph-Evolution Learning," in IEEE Transactions on Parallel and Distributed Systems, vol. 35, no. 3, pp. 499-516, March 2024, doi: 10.1109/TPDS.2024.3357715. keywords: {Dynamic scheduling;Heuristic algorithms;Prediction algorithms;Correlation;Predictive models;Interference;Graph neural networks;Workload prediction;cloud computing;graph neural network;resource management;datacenter},

[14] D. Alqahtani, "Leveraging Sparse Auto-Encoding and Dynamic Learning Rate for Efficient Cloud Workloads Prediction," in IEEE Access, vol. 11, pp. 64586-64599, 2023, doi: 10.1109/ACCESS.2023.3289884.
keywords: {Cloud computing;Logic gates;Deep learning;Quality of service;Data centers;Predictive models;Forecasting;Cloud computing;deep learning;sparse auto-encoder;gated recurrent unit;learning rate},

[81] Takale, Dattatray & Wattamwar, Aditya & Saipatwar, Saksham & Saindane, Harshwardhan & Patil, Tushar. (2024). Comparative Analysis of LSTM, RNN, CNN and MLP Machine Learning Algorithms for Stock Value Prediction. 2. 1-10. 10.48001/JoFSN.

[82] M. D. P. P. Goonathilake and P. P. N. V. Kumara, "CNN, RNN-LSTM Based Hybrid Approach to Detect State-of-the-Art Stance-Based Fake News on Social Media," 2020 20th International Conference on Advances in ICT for Emerging Regions (ICTer), Colombo, Sri Lanka, 2020, pp. 23-28, doi: 10.1109/ICTer51097.2020.9325477.

[16] Ahamed, Z., Khemakhem, M., Eassa, F., Alsolami, F., Basuhail, A., Jambi, K. Deep Reinforcement Learning for Workload Prediction in Federated Cloud Environments. Sensors 2023, 23, 6911. https://doi.org/10.3390/s23156911

[17] AHAMED, Z. et al. Technical Study of Deep Learning in Cloud Computing for Accurate Workload Prediction. Electronics (2079-9292), [s. l.], v. 12, n. 3, p. 650, 2023. DOI 10.3390/electronics12030650. Disponível em: https://research.ebsco.com/linkprocessor/plink?id=9ac0e5da-5a35-30c7-b3b7-4d786019fe7e. Acesso em: 28 ago. 2024.