Explainable Task Failure Prediction in Cloud Datacenter Using Machine Learning

Afsana Kabir Sinthia
ID: 23166004
Department of Computer Science and Engineering
Brac University
afsana.kabir.sinthi@g.bracu.ac.bd

Sadia Islam
ID: 23166021
Department of Computer Science and Engineering
Brac University
sadia.islam15@g.bracu.ac.bd

Iffat Ara Jui
ID: 23166010
Department of Computer Science and Engineering
Brac University
iffat.ara.jui@g.bracu.ac.bd

Nisat Islam Mozumder
ID: 22366047
Department of Computer Science and Engineering
Brac University
nisat.islam.mozumder@g.bracu.ac.bd

Md. Tariqul Islam
ID: 23173006
Department of Computer Science and Engineering
Brac University
md.tariqul.islam5@g.bracu.ac.bd

Ehsanur Rahman Rhythm
Department of Computer Science and Engineering
Brac University

Sania Azhmee Bhuiyan
Department of Computer Science and Engineering
Brac University

***Abstract*—A new strategy is needed to increase the dependability and availability of cloud services for contemporary applications like smart cities, home automation, and eHealth. Due to the cloud environment's vastness and variety, most cloud services, including hardware and software, have failed. Using publicly accessible traces, we first analyze and characterize the behaviour of failed and successful tasks in this study. We have designed and developed a failure prediction model in order to anticipate task failures. The proposed model seeks to improve cloud application efficiency and resource consumption. We evaluate the proposed model using publicly available traces: the Alibaba cluster. In addition, the traces were subjected to a variety of machine learning models to determine the most precise one. Our findings demonstrate a correlation between unsuccessful assignments and requested resources. The evaluation results also demonstrated that our model possesses high accuracy, recall, and F1 scores with explainable AI. Solutions, including the prediction of job failure, can enhance the dependability and availability of cloud services.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

Shared clusters have emerged as a preferred infrastructure solution for hosting large, protracted applications as a result of the cloud computing industry's quick development and increasing demand for scalable and effective services. Thus, the allocation of resources among applications is currently coordinated by a central resource manager, while dedicated application managers oversee application-specific operations. [6].

The deployment of cloud systems has opened avenues to address a range of cloud-related failures [12]. In order to effectively manage interference between coexisting workloads and isolate resources, solutions for workload co-location have been explored [7]. By combining many services into a single shared cluster, resource usage is optimized, leading to significant cost savings.

Yet, managing persistent services within shared clusters poses challenges.It's critical for service providers to deploy resources efficiently in order to meet performance objectives while minimizing resource waste. Proactive resource provisioning and capacity planning depend on anticipating workload patterns and resource requirements for ongoing services.

To optimize resource allocation and ensure dependable service delivery, novel approaches and algorithms are vital for workload management and forecasting in shared clusters. It encompasses workload classification, performance prediction, resource scheduling, and load balancing.

This study aims to propose effective methods for managing and forecasting workloads in shared clusters that support long-lasting services. Leveraging historical workload data, machine learning, and optimization, intelligent models and algorithms can enhance resource allocation, workload prediction, and proactive capacity planning.

The study clarifies the opportunities and difficulties in managing enduring services within shared clusters, so enhancing overall performance and dependability. By creating precise workload forecasting models and clever resource allocation strategies, this study responds to the changing requirements of current applications.

In the face of hardware and software failures in the expansive cloud environment, machine learning has emerged as a tool to understand complex dynamics of job failures.

An analysis of failed and successful tasks, using publicly accessible traces, forms the foundation for a novel failure prediction model. This predictive framework optimizes cloud application efficiency and resource usage.

The model's efficacy is validated through recise analysis of traces from the Alibaba cluster [4]. A correlation between requested resources and task success is identified. This model, enriched with explainable AI, offers high accuracy, recall, and F1 scores.

By embracing machine learning, this proposed failure prediction model has the potential to revolutionize cloud services, going beyond prediction to enhance dependability. In this context, this paper delves into the realm of task failure prediction, building upon prior works' insights [2]–[4], [15]. Through machine learning, it aspires to unveil task failure complexities and contribute to a more dependable cloud datacenter environment.

II. Literature Review

Making an accurate prediction for cloud workload and accurately analyzing the task failure data is essential to effectively managing cloud resources and enhancing service quality. The majority of earlier works were on simple RNN and LSTM methods which face the vanishing gradient problem vastly and cannot provide much accurate prediction.

For this reason, the paper [1] brought out a spectacular idea to improve the forecast accuracy of a large shared cluster dataset. The authors proposed an ensembled forecasting module for higher prediction accuracy where the VMD decreases the randomness of workload sequences and the LocalRNN obtains the local non-linear relationship and the R-Transformer obtains nonlinear information of time series.

According to the paper [2] they propose a failure prediction model that can detect failure early before it suddenly occurs. Moreover, they analyze failure behavior and study various cloud traces. The paper [3] talks about a distributed file system HDFS and a framework that analyzes and transforms very large data sets using the MapReduce paradigm.

Another research paper [4] proposes a failure prediction algorithm based on multi-layer BiLSTM to identify task and job failures in cloud data centers.It achieved around 93 percent accuracy for task failure prediction.In the paper [5] the authors built a model which uses a hybrid GA-PSO algorithm to train a functional link neural network (FLNN) for higher workload prediction accuracy.

This research [8] proposes a failure-aware task-scheduling structure that applies ANN and CNN to predict the state of given tasks' termination during runtime. They used the ILP model for the action selection problem. The deep learning models gave up to 94 percent failure prediction accuracy and the heuristic optimization technic helped to save 40 percent of resource usage.

Some researchers [9] used Adaboost ensemble classifier using Regression, Random Forest, and Decision Tree algorithms to predict cloud failure with higher accuracy whereas some other researcher [10] used just an enhanced LSTM method and got quite good accuracy score. Moreover, the paper [11] used ML classifiers to detect failed jobs before the cloud management system schedules them.

According to new research [13] a model has been proposed based on online sequential extreme learning machine (OS-ELM) to predict online job termination status where real-time streaming data are collected according to the sequence of job arrival. By intelligently recognizing failed jobs, it lowers the storage space overhead and drastically lowers cloud resource waste.Another paper [12] proposed a performance comparison and evaluation among five ML and 3 deep learning models which were built and trained to predict the job and task termination status which includes logistic regression, decision tree, random forest, gradient boost and XGBoost classifiers and LSTM.

A popular research [16] showed a model for a fault prediction system in cloud infrastructure. This model consists of a cloud infrastructure management unit and a ML based fault prediction unit. The orchestrator from the first unit has the ability to manage the VM-based cloud infrastructure and container-based cloud infrastructure.The fault prediction system comprises of a receiving unit that receives monitoring data, a data processing unit to process it, and a fault prediction system that analyzes the fault prediction values.

III. Methodology

The primary elements of the suggested framework for workload analysis and prediction in the cloud are covered in this section.The implemented workload prediction model has been designed to anticipate the outcome of submitted tasks prior to their execution.

The method accepts a workload made up of a number of jobs called a cloud trace workload, designated as D. The chosen cloud trace is then subjected to a variety of feature selection strategies and classifier models by the algorithm. It assesses how well these methods and models perform. The result of the algorithm indicates whether the termination was "failed" or "finished" respectively. The subset of data that was taken from the input cloud workload trace here we used Alibaba cloud trace is represented by the dataset D. Cleaning and filtering operations are used to eliminate jobs that have been submitted in excess or stopped because they are resource-intensive as part of the pre-processing of the data. Both the training and testing of the prediction models use the chosen cloud trace. Using the prediction model M, tasks are classified as either failed or finished.

The suggested model's procedure may be summed up as follows:

(1) Different traces were collected for the purpose of ensuring the applicability of our model to traces of varying lengths.

(2) To prepare the data for classification and modeling, analysis, preprocessing, and filtering techniques were applied to input traces.

(3) The traces underwent three feature selection techniques to enhance the accuracy and performance of the proposed

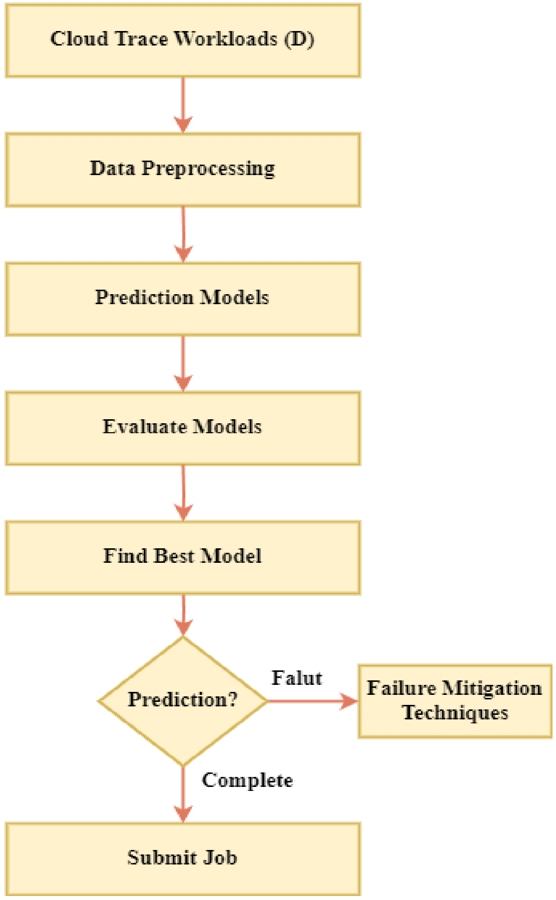

Fig. 1: The proposed evaluation process

model. Subsequently, the most significant features were ranked based on the obtained results.

(4) Than use machine learning classification approaches were utilized on the traces to predict failed and finished jobs.

(5) Ultimately, based on the best prediction outcomes, the cloud management system determines the appropriate failure prediction model.

If a job is predicted as "finished" it is submitted and typically scheduled on available nodes. In the case of an incoming task being predicted as "failed" future work will address the implementation of failure mitigation techniques.

The main objective of this prediction model is to accurately and early predict the status of tasks (whether "failed" or "finished") in cloud applications using machine learning classification algorithms. By implementing the proposed model, computational time and resource usage are reduced, while simultaneously enhancing the efficiency and performance of the cloud infrastructure.

IV. Background

Task Failure Prediction and Resource allocation is the allocation of resources and services from a cloud provider to users. It is the process of choosing, deploying, managing software and hardware to ensure application performance.

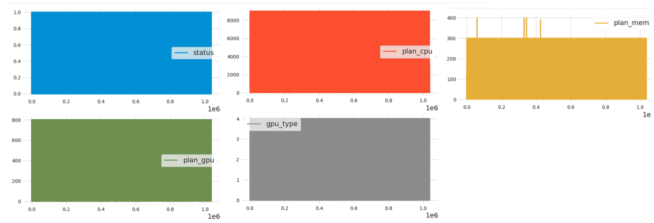

Fig. 2: Analysis of Alibaba cloud trace dataset

There are four types of resource allocation strategies: (1)Dynamic (2)Linear Scheduling (3)Particle Swarm Optimization (4)ACO algorithm. Dynamic resource allocation is generally used for load balancing. In this method loads are distributed among Virtual Machines(VMs). Linear Scheduling maximizes the system through put and resource use. ACO(ant colony optimization) algorithm solves load balancing problems. It helps in achieving better resource usage and higher throughput. Workload prediction is used to predict information for future. Forecasting takes information available in the present and uses it to predict the future. This can improve efficiency and reduce the operational cost of the cloud. Proactive capacity planning includes utilizing the network, production capacity and storage capacity management tools to predict network, production and storage needs. It also implement preemptive, corrective actions. Optimization algorithms are using in this model to reach these results. They are using for minimizing the error, making predictions on data, learning from the training data sets, classifying the task and regression the task. Here we have used KNN, Xgboost, Random Forest and SVM algorithms for training our dataset.

KNN: KNN(K-Nearest Neighbors) is a supervised learning algorithm. KNN predicts the correct class for the test dataset. It calculated the distance between the test data and all the training points. Then it selects the K number of point which is very close to the test data. In our work Scikit-Learn's K Nearest Neighbors Classifier's prediction accuracy is: 78.91 Time consumed for training: 2.590 seconds Time consumed for prediction: 658.40658 seconds.

Xgboost: Xgboost is used for supervised learning as a gradiant boosting algorithm. It is highly efficient and scalable implementation. Here XGBoost's prediction accuracy is: 79.52 Time consumed for training: 38.288 Time consumed for prediction: 0.59685 seconds

Random Forest: Random Forest algorithm is an ensemble learning technique. It combines numerous classifier to enhance model's performance. It contains several decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. In our work Scikit-Learn's Random Forest Classifier's prediction accuracy is: 79.64 Time consumed for training: 6.088 seconds Time consumed for prediction: 0.38745 seconds.

SVM: SVMs are adaptable and efficient in a variety of applications. They can manage high dimensional data and non-linear relationship.

V. Performance Evaluation

In this section, we evaluate our trained models and compare their performance with each other. We reviewed the outcomes of seven models that were developed and tested by calculating their accuracy, Time consumed for training and Time consumed for prediction value and using unknown data. Since temporal data points are depicted on a single row (typically rendered as rectangular cells and color-coded by their values), the heatmap is the most effective method for managing a large number of time series. Nonetheless, as the number of observations increases, the expense of rendering individual map cells in the context of the big dataset rises. To mitigate this issue while retaining the benefit of the heatmap depiction, we displayed a continuous heatmap for displaying our time series data, which enabled users to identify similar time series patterns by positioning similar time series closer together. The visual layout of this map is depicted in Fig. 3. Specifically, the colour represents the CPU utilization of each machine (maximum 100%). The continuous heatmap significantly reduces rendering time because the number of continuous areas is substantially less than the number of actual heatmap cells.

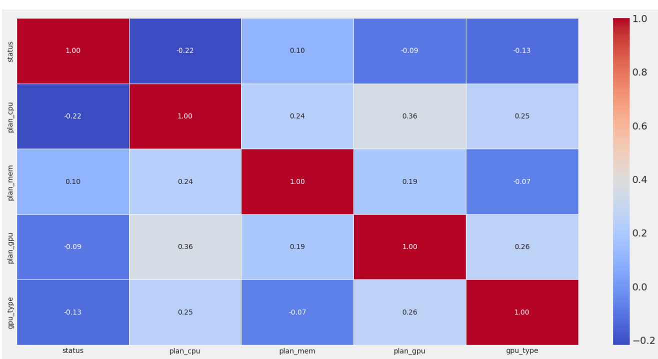

Fig. 3: Heatmap visualization of the correlation matrix

TABLE I: Performance Measures

Models	Time consumed for training (sec)	Time consumed for prediction (sec)
KNN	2.590	0.65840
XGBoost	3.828	0.59685
RandomForest	6.088	0.38745
SVM	4.098	0.27854

VI. Conclusion

References

[1] S. Zhou, J. Li, K. Zhang, M. Wen and Q. Guan, "An Accurate Ensemble Forecasting Approach for Highly Dynamic Cloud Workload With VMD and R-Transformer," in IEEE Access, vol. 8, pp. 115992-116003, 2020, doi: 10.1109/ACCESS.2020.3004370.

[2] Jassas, M. S., Mahmoud, Q. H. (2022). Analysis of Job Failure and Prediction Model for Cloud Computing Using Machine Learning. Sensors, 22(5), 2035. https://doi.org/10.3390/s22052035.

[3] The Hadoop Distributed File System Konstantin Shvachko, Hairong Kuang, Sanjay Radia, Robert Chansler Yahoo! Sunnyvale, California USA Shv, Hairong, SRadia, Chansler@Yahoo-Inc.com .

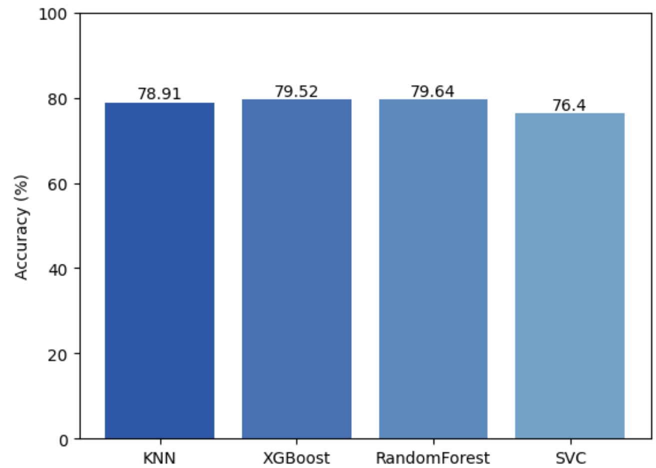

Fig. 4: Accuracy of all the models

[4] J. Gao, H. Wang and H. Shen, "Task Failure Prediction in Cloud Data Centers Using Deep Learning," in IEEE Transactions on Services Computing, vol. 15, no. 3, pp. 1411-1422, 1 May-June 2022, doi: 10.1109/TSC.2020.2993728.

[5] Malik, S. Z., Tahir, M., Sardaraz, M., Alourani, A. (2022). A Resource Utilization Prediction Model for Cloud Data Centers Using Evolutionary Algorithms and Machine Learning Techniques. Applied Sciences, 12(4), 2160. https://doi.org/10.3390/app12042160.

[6] Zhu, J., Yang, R., Sun, X., Wo, T., Hu, C., Peng, H., Xiao, J., Zomaya, A.Y. and Xu, J., 2022. QoS-aware co-scheduling for distributed long-running applications on shared clusters. IEEE Transactions on Parallel and Distributed Systems, 33(12), pp.4818-4834.

[7] P. Lama, S. Wang, X. Zhou, and D. Cheng, "Performance isolation of data-intensive scale-out applications in a multi-tenant cloud," in Proc. IEEE Int. Parallel Distrib. Process. Symp., 2018, pp. 85–94

[8] Alahmad, Y., Daradkeh, T., Agarwal, A. (2021). Proactive failure-aware task scheduling framework for cloud computing. IEEE Access: Practical Innovations, Open Solutions, 9, 106152–106168. https://doi.org/10.1109/access.2021.3101147

[9] Ng'ang'a, D. N., Cheruiyot, W. K., Njagi, D. (2023). A machine learning framework for predicting failures in cloud data centers -A case of Google cluster -azure clouds and alibaba clouds. https://doi.org/10.2139/ssrn.4404569

[10] Islam, T., Manivannan, D. (2017). Predicting application failure in cloud: A machine learning approach. 2017 IEEE International Conference on Cognitive Computing (ICCC).

[11] Jassas, M. S., Mahmoud, Q. H. (2020). Evaluation of a failure prediction model for large scale cloud applications. In Advances in Artificial Intelligence (pp. 321–327). Springer International Publishing.

[12] Tengku Asmawi, T.N., Ismail, A., Shen, J. Cloud failure prediction based on traditional machine learning and deep learning. J Cloud Comp 11, 47 (2022). https://doi.org/10.1186/s13677-022-00327-0

[13] Liu, C., Han, J., Shang, Y., Liu, C., Cheng, B., Chen, J. (2017). Predicting of job failure in compute cloud based on online extreme learning machine: A comparative study. IEEE Access: Practical Innovations, Open Solutions, 5, 9359–9368. https://doi.org/10.1109/access.2017.2706740

[14] Chen, X., Lu, C.-D., Pattabiraman, K. (2014). Failure analysis of jobs in compute clouds: A Google cluster case study. 2014 IEEE 25th International Symposium on Software Reliability Engineering.

[15] Yoo, W., Sim, A., Wu, K. (2016). Machine learning based job status prediction in scientific clusters. 2016 SAI Computing Conference (SAI).

[16] Yang H, Kim Y. Design and Implementation of Machine Learning-Based Fault Prediction System in Cloud Infrastructure. Electronics. 2022;11(22):3765.https://doi.org/10.3390/electronics11223765

[17] Predicting Job Failure in Cloud Cluster: Based on SVM Classification. (2016). JOURNAL OF BEIJING UNIVERSITY OF POSTS AND TELECOM, 39(5), 104–109.

[18] Saxena, D., Kumar, J., Singh, A. K., Schmid, S. (2023). Performance

analysis of machine learning centered workload prediction models for cloud. In arXiv [cs.DC]. http://arxiv.org/abs/2302.02452