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*—There is a pressing need for the development of a novel approach aimed at enhancing the reliability and accessibility of cloud services in order to cater to the requirements of modern applications such as smart cities, home automation, and eHealth. The failure of numerous cloud services, encompassing both hardware and software, can be attributed to the expansive and diverse nature of the cloud environment. Using publicly accessible traces, we first analyze and characterize the behavior 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 accessible traces: the Alibaba cluster. Furthermore, the traces underwent analysis using a range of machine-learning models in order to ascertain the model that yielded the highest level of accuracy. Our findings demonstrate a significant association between unsuccessful assignments and the extent of resource utilization. The evaluation findings further indicated that our model possesses high accuracy. 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 model's efficacy is validated through precise 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 data center environment.

We have suggested a system that analyzes cloud trace failure. Based on the available knowledge, it appears that only a limited number of prior studies have developed and executed a failure prediction model utilizing various classifiers derived from machine learning techniques. These studies have also applied the proposed model to Alibaba cloud traces. The evaluation of the model's performance has been conducted using multiple criteria to ascertain the ability of the proposed model to achieve a high level of accuracy during the prediction phase. Hence, the primary contribution of this study might be summarized as follows:

- To analyzes several cloud traces to determine the relationship between job/task failure and cloud trace attributes.
- Develop and execute a failure prediction model combining several machine learning techniques with cloud workload traces.
- In order to ascertain the model's generality and applicability to diverse cloud trails, it is vital to conduct an evaluation of our proposed model.

The remainder of the paper is organized as follows: Section 2 provides a comprehensive review of existing approaches and research efforts in long-running service management and workload forecasting. Section 3 presents the proposed methodology, encompassing workload characterization, performance prediction, resource scheduling, and load balancing. Section 4 describes the experimental setup, presents the results, and provides an analysis. Finally, Section 5 concludes the paper, summarizing the findings and proposing future research directions.

II. Literature Review

The solution to efficiently managing cloud resources and ensuring service quality is precisely estimating cloud workload and analyzing task failure data. Most past research focused on basic RNN and LSTM techniques, which suffer greatly from the vanishing gradient problem and cannot make reliable predictions.

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

According to the paper [2], they propose an approach for predicting failure that may detect failure before it occurs. Additionally, they analyze multiple cloud traces and failure behavior. The paper [3] talks about a distributed file system HDFS and a framework that analyzes and transforms large data sets using the MapReduce paradigm. Another paper [15] worked on managing the huge volume of logs for training and prediction by parallelizing the log analysis.

Another research paper [4] provides a failure prediction algorithm based on the multi-layer BiLSTM to find failure jobs and tasks in cloud data centers. It achieved around ninety-three percent accuracy for task failure prediction. In the paper [5] the authors of the research developed a model that trains a functional link neural network (FLNN) using a hybrid GA-PSO method for more accurate workload prediction.

This research [8] suggests a failure-aware task-scheduling framework that uses ANN and CNN to predict the outcome of given tasks terminating state in the runtime. They used the ILP model for the action selection problem. The deep learning models gave up to ninety-four percent failure prediction accuracy and the heuristic approach helped to save 40 percent of resource usage.

Some researchers [9] used the 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 a good accuracy score. Moreover, the paper [11] applied ML classifiers to detect failing jobs before the cloud management system scheduled them.

According to new research [13], real-time streaming data is collected in accordance with the order of job arrival in a model based on an online sequential extreme learning machine that forecasts online job termination status. 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 LSTM, logistic regression, decision trees, random forests, gradient boost, and XGBoost classifiers which were built and trained are used to forecast the job and task termination status.

In a research [17] SVM based model has been proposed to forecast the termination status of jobs where combine of static and dynamic characteristics are selected as feature vectors. The model performed quite better than the traditional neural network model. According to paper [18] the study compares five diverse machine learning-driven cloud workload prediction models through performance analysis. The evaluation focuses on overcoming difficulties including inconsistent resource capacity, varied priorities, the uncertainty of resource demand over time, data granularity, prediction window size, and assessing the constraints and findings of the models.

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 an ML-based fault prediction unit. Both the container-based and VM-based cloud architecture can be handled by the orchestrator from the first unit. A receiving device that receives monitoring data, a data processing unit to process it, and a fault prediction system that analyzes the fault prediction values comprise the fault prediction system.

III. 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, and managing software and hardware to ensure application performance. Workload prediction is used to predict information for the 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 implements preemptive, corrective actions. Optimization algorithms are used in this model to reach these results. They are used for minimizing the error, making predictions on data, learning from the training data sets, classifying the task, and regression of the task. Cloud management system generally faces different types of cloud failures. When the system works on the implementation and deployment failure occurs most of the time. It is very important to find the failures. If the system is unable to handle failures, service quality, system availability, and reliability will fall apart. Failure in the cloud: Generally when a system cannot perform its modeled function because of hardware dysfunction or a sudden halt of the software then the cloud computing system fails.

There are two types of failures that are common in cloud computing (a) Structural Failures and (b) Situational Failures. Structural failures include performance and resource failures and situational failures include dependent failures and independent failures. To solve these problems task failure management is needed. In this management, there are some categories of fault management such as: stop redundancy, fault prediction, and load balancing. In this paper, we have tried to establish a model that can predict failure using machine learning algorithms. Here we have used KNN, Xgboost, Random Forest, and SVM algorithms for training our dataset.

KNN: An algorithm for supervised learning is KNN (K-Nearest Neighbors). For the test dataset, KNN accurately predicts the correct class. It determined the separation between all of the training points and the test data. Then it chooses K points that are relatively close to the test data.

Xgboost: As a gradient-boosting approach for supervised learning, Xgboost is utilized. Its implementation is both highly effective and scalable.

Random Forest: A method used in ensemble learning is the Random Forest algorithm. It combines several classifiers to improve the performance of the model. In order to increase the dataset's predictive accuracy, it contains many decision trees on different subsets of the given dataset and takes the average.

SVM: Support Vector Machines (SVMs) exhibit adaptability and efficiency across a diverse range of applications. The ability to effectively handle data with a large number of dimensions and non-linear relationships is within their capacity.

IV. Methodology

In Methodology part outlines the key components of the implied methodology for workload analysis and prediction in the cloud. The workload prediction model in use has been created to foresee how submitted tasks will turn out before they are completed.

The method accepts a workload made up of a number of jobs called a cloud trace workload, designated as D. The method then applies a number of methods for feature selection and classifier models on the chosen cloud trace. It evaluates how effectively these techniques and models work 1. The algorithm's output reveals whether the termination was "failed" or "finished" in turn. The dataset D is the subset of data that was collected from the input cloud workload trace; in this case, we utilized Alibaba cloud trace. As part of the pre-processing of the data, cleaning and filtering activities are performed to remove jobs that have been submitted in excess or halted because they need a lot of resources. Both the training and testing of the prediction models employ the chosen cloud trace. The prediction model M is used to categorize tasks as either completed or unsuccessful.

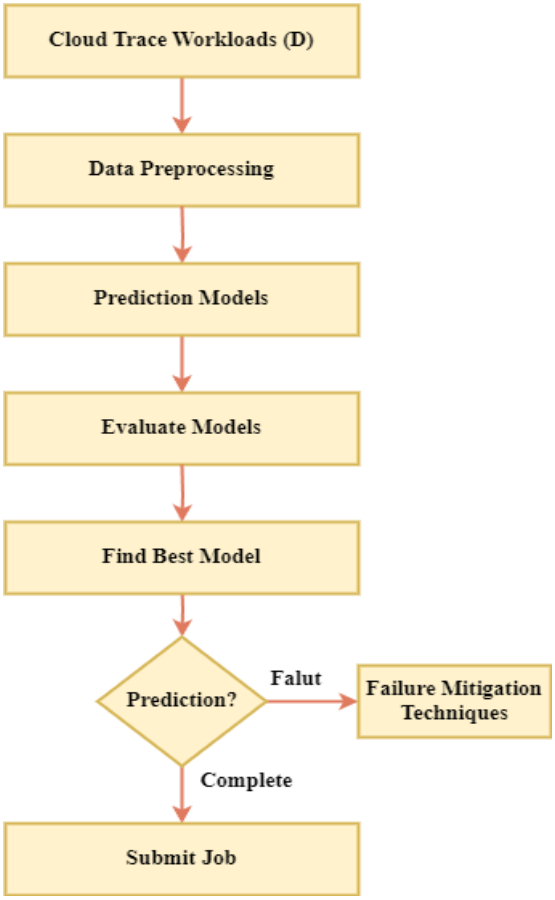

Fig. 1: The proposed evaluation process

The process of the proposed model may be summed up as follows:

- A variety of traces of different lengths might be used to apply our approach.
- Input traces were subjected to analysis, pre-processing, and filtering processes in order to get the data ready for

classification and modeling.

- To improve the precision and effectiveness of the suggested model, the traces were subjected to three feature selection procedures. On the basis of the outcomes, the most important traits were then rated.
- Afterward, machine learning classification methods were employed on the trace data to make predictions regarding both unsuccessful and successful employment outcomes.
- Finally, the cloud management system selects the most optimal failure prediction model by evaluating the outcomes of the most accurate predictions.

A project is submitted and normally scheduled on available nodes if it is anticipated to be ”finished.” Future study will focus on the application of failure mitigation approaches when an incoming task is likely to be ”failed.”

The major goal of this prediction model is to use machine learning classification techniques to reliably and promptly anticipate the status of jobs (whether ”failed” or ”finished”) in cloud applications. By putting the suggested approach into practice, computational time and resource utilization are cut while the cloud infrastructure's effectiveness and performance are improved.

1) Data Preprocessing and Filtering:: A set of features that may define the job/task attributes and the behavior of a cloud system make up the model's input. We have work datasets from Alibaba's cloud trace, and these jobs and tasks are displayed with a time of 0. Additionally, we changed the timestamp of every work and job to be in the daylight. We looked into why there were so many unsuccessful tasks in Alibaba Cloud Trace compared to failed jobs, and we discovered that some tasks had to be submitted thousands of times before they could be properly completed. These efforts, nevertheless, were terminated after using up a significant amount of resources. Because they are regarded as outlier situations, we have eliminated certain jobs, which include these sorts of activities.

2) Prediction Techniques:: In our study, one of the supervised learning algorithms we use is the Decision Trees (DTs) classifier. In this research, we used four Machine Learning (ML) classification algorithms: Random Forest (RF), K-Nearest Neighbours (KNN), XGBoost, and Support vector machine (SVM).

V. Performance Evaluation

In this section, we assess our trained models' performance and contrast it with that of other models. By assessing the accuracy, time spent on training, time spent on prediction value, and using unreliable data, we evaluated the results of seven constructed and tested models.

A study examines the merits and downsides of managing enduring services in shared clusters to improve performance and reliability. Accurate workload forecasting and clever resource allocation help this study address changing application needs. In the developing cloud environment, hardware and software concerns have led to machine learning for job failures. A new failure prediction model uses public traces to examine failed and successful activity. Predictive framework optimizes cloud application resources. In Figure 2, we have conducted an analysis of the classes of Alibaba datasets, with a particular focus on the original Time Series. The data was transformed into binary format to represent success and failure across five classes: $status$, $plan-cpu$, $plan-mem$, $plan-gpu$, and $gpu-type$.

The heatmap is the most efficient way to manage a lot of time series since temporal data points are displayed on a single row (usually rendered as rectangular cells and color-coded by their values). Nevertheless, the cost of drawing a single map cell in the context of a large dataset rises as the number of observations does. We exhibited a continuous heatmap for displaying our time series data in order to address this problem while maintaining the advantages of the heatmap portrayal. This allowed users to recognize patterns in comparable time series by grouping similar time series closer together. Fig. 3 heatmap shows how this map's visual structure is organized. The color specifically denotes the CPU usage of each computer (up to 100%). Because there are far fewer continuous areas than there are actual heatmap cells, the continuous heatmap drastically lowers rendering time.

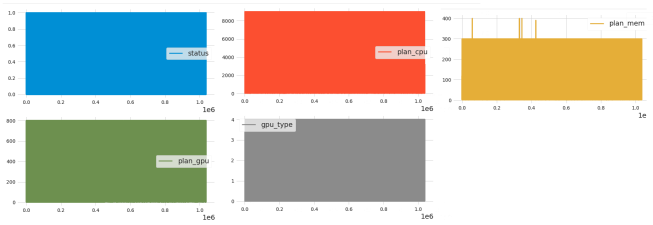

Fig. 2: Analysis of Alibaba cloud trace dataset

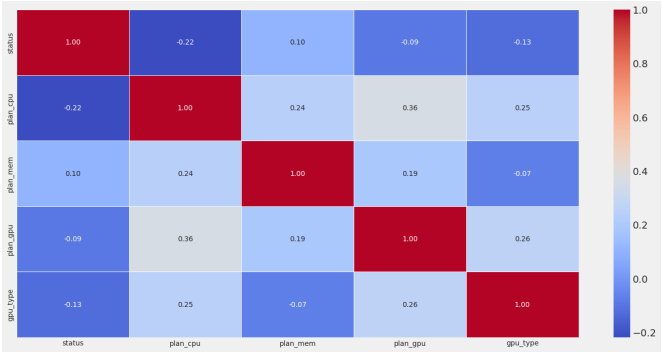

Fig. 3: Heatmap visualization of the correlation matrix

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

Table I presents the performance metrics of each model when trained on the training dataset, specifically focusing on

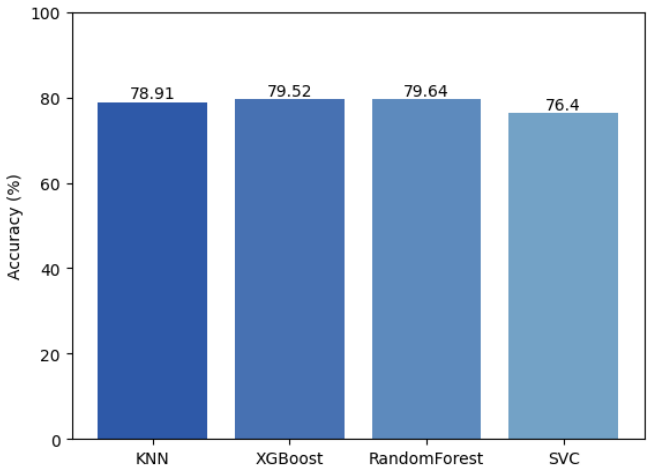

Fig. 4: Accuracy of all the models

the time required for training and prediction. On the other hand, Figure 4 illustrates the performance of all models trained to classify the termination status of the test dataset. It is evident that the performance of the Support Vector Classifier (SVC) and K-Nearest Neighbors (KNN) is comparatively inferior to that of XGBoost and Random Forest. The Random Forest model demonstrates the highest level of accuracy, at a rate of 79.64%. The training time required for this task is 6.088, but the prediction time is significantly lower at 0.38755. This suggests that the model is capable of accurately predicting the job termination status. The evaluation of other studies indicates that the K Nearest Neighbors Classifier in Scikit-Learn achieves a prediction accuracy of 78.91%. The training process takes approximately 2.590 seconds, while the prediction process takes approximately 658.40658 seconds. The predictive accuracy of XgBoost in this particular scenario is 79.52%. A total of 38.288 minutes were allocated for the training session. The duration of the prediction process was 0.59685 seconds.

VI. Conclusion

Failed tasks consume significant resources within cloud clusters, including computing time, CPU utilization, RAM, and disk space. The analysis of these failures characterizes the behaviors of both unsuccessful and completed tasks, aiming to establish connections between failed tasks and other attributes of cloud applications. The primary goal is to predict impending task failures. This proposed model finds applicability in extensive data centers and cloud computing environments, where it can anticipate task failures ahead of scheduling by the cloud management system. The creation and implementation of a failure prediction model offer advantages such as heightened performance and reduced expenses associated with cloud utilization. To this end, diverse classification algorithms have been employed across various workload traces to formulate a comprehensive model that excels at accurately foretelling task failures with notable precision. Multiple feature selection techniques have also been harnessed to enhance the accuracy of the failure prediction model. Remarkably, our approach has achieved exceptional accuracy for Alibaba trace, leveraging RF classifier in conjunction with RFE feature selection, yielding a remarkable accuracy of 79.64%. Looking ahead, our future endeavors include exploring the application of deep learning models to publicly available traces, along with a more comprehensive exploration of mitigation strategies and methodologies.

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 .

[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] Ahmad, 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