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 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 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 become a well-liked infrastructure model for hosting numerous long-running services as a result of the rapid expansion of cloud computing and the rising demand for scalable and cost-effective services. Multiple services from various tenants or users are consolidated into a single pool of resources in shared clusters, resulting in effective utilization of resources and cost savings.

However, there are substantial difficulties in managing persistent services in shared clusters. The performance criteria of each service must be met, and resource waste must be minimized, according to service providers, who must make sure that the resources are allocated effectively. Additionally, for proactive resource provisioning and capacity planning, anticipating the workload patterns and resource requirements of long-running services is essential.

In order to optimize resource allocation and guarantee dependable service delivery, new approaches and algorithms are needed for workload management and forecasting in shared clusters. This requires taking care of several issues, including workload classification, performance prediction, resource scheduling, and load balancing.

This study's goal is to investigate and suggest effective methods for managing and forecasting the workload in shared clusters that support long-lasting services.By leveraging historical workload data, machine learning techniques, and optimization algorithms, we can develop intelligent models and algorithms that enable efficient resource allocation, workload prediction, and proactive capacity planning.

This study will advance the discipline by shedding light on the potential and difficulties associated with managing enduring services in shared clusters.The overall performance, dependability, and cost-effectiveness of shared cluster systems can be enhanced by creating precise workload forecasting models and smart resource allocation procedures.

The overall goal of this research is to address the difficulties in managing and forecasting workload in shared clusters hosting long-running applications, allowing service providers to efficiently allocate resources, boost performance, and guarantee dependable service delivery in shared cluster environments.

II. Literature Review

Making an accurate prediction for cloud workload is essential to effectively managing cloud resources, enhancing service quality, and preventing Service-Level Agreement agreements from being violated. The majority of earlier works on cloud workload predictions were built using recurrent neural networks (RNN). However, these RNN-based methods are insufficient in obtaining the linear and non-linear relationships and cannot provide accurate forecasts in a highly dynamic cloud workload scenario where resource utilization changes faster and more frequently. This is because classic RNN suffers from the issue of vanishing gradient.

For this reason, the paper [1] brought out a spectecular idea to improve the forecast accuracy of large shared cluster dataset.The authors here proposed an ensembled forecasting module based on R-transformer and auto regressive model and Variational Mode Decomposition. They Pre process workload data sequence and decompose it into multiple Intrinsic Mode Functions (IMF) using VMD To decrease the randomness of highly dynamic cloud workload sequences. After preprocessed IMFs inputed to the ensemble module it uses LocalRNN to obtain the local non-linear relationship .

The multi-head attention mechanism of R-Transformer is used to obtain local and global non linear information of time series to achieve higher prediction accuracy and helps to captures long-term dependencies in various sequence modeling task. The auto regressive model deals with the linear relationship of workload.

The reliability and availability of cloud applications must be improved for contemporary applications like smart home automation, and eHealth. The immense size and variety within the cloud environment has resulted in inefficient distribution management regulations and algorithms to cope up with big data environment.

According to the paper [2] they propose a model for failure prediction that can detect failure early before it suddenly occurs. Moreover, they analyze failure behavior and study various cloud traces in Google cluster, Mustang and Trinity traces to find the correlation between (failed and successful) jobs/tasks by implementing machine learning algorithm.The study also found that long-running jobs consumed more resources than finished jobs although high-priority jobs were more likely to fail.

The paper [3] talks about a distributed file system HDFS and a framework which analyze and transform very large data sets using the MapReduce paradigm.It is designed on principle of write-once and read-many-times. Once data is written large portions of dataset can be processed any number times. Moreover, the Hadoop Distributed File System (HDFC) can do computations across multiple (thousands) of hosts, partition data, and run application computations in parallel.HDFS Client is a code library that exports the HDFS file system interface.User applications can access the file system using the HDFS client.The HDFS is used to manage 25 petabyte of enterprise data at Yahoo.

In same way, another research paper [4] proposes a failure prediction algorithm based on multi-layer Bidirectional Long Short Term Memory (BiLSTM) to identify task and job failures in cloud data centers.It achieved around 93 percent accuracy for task failure prediction and 87 percent accuracy for job failure prediction but has some major limitations.THere is a paper [5] where the authors built a model which uses a hybrid Genetic Algorithm-Particle Swarm Optimization (GA-PSO) algorithm to train a functional link neural network (FLNN). The FLNN is then used to predict resource utilization for future time intervals.The prediction accuracy was signifantly high withtheir machine learning t

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 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.

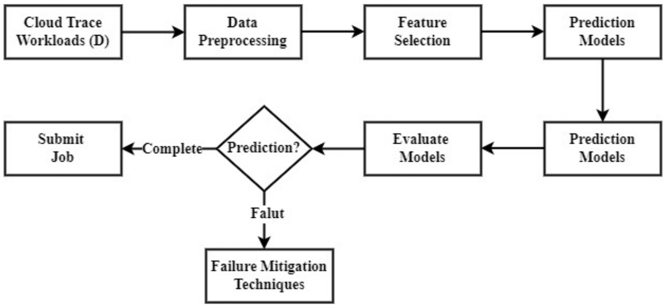

Fig. 1: The proposed evaluation process

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 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

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. 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.

V. Prepare Your Paper Before Styling

Before you begin to format your paper, first write and save the content as a separate text file. Complete all content and organizational editing before formatting. Please note sections V-A–V-E below for more information on proofreading, spelling and grammar.

Keep your text and graphic files separate until after the text has been formatted and styled. Do not number text heads—LaTeX will do that for you.

A. Abbreviations and Acronyms

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, ac, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

B. Units

- Use either SI (MKS) or CGS as primary units. (SI units are encouraged.) English units may be used as secondary units (in parentheses). An exception would be the use of English units as identifiers in trade, such as “3.5-inch disk drive”.
- Avoid combining SI and CGS units, such as current in amperes and magnetic field in oersteds. This often leads to confusion because equations do not balance dimensionally. If you must use mixed units, clearly state the units for each quantity that you use in an equation.
- Do not mix complete spellings and abbreviations of units: “Wb/m^2” or “webers per square meter”, not “webers/m^2”. Spell out units when they appear in text: “. . . a few henries”, not “. . . a few H”.
- Use a zero before decimal points: “0.25”, not “.25”. Use “cm^3”, not “cc”.)

C. Equations

Number equations consecutively. To make your equations more compact, you may use the solidus (/), the exp function, or appropriate exponents. Italicize Roman symbols for quantities and variables, but not Greek symbols. Use a long dash rather than a hyphen for a minus sign. Punctuate equations with commas or periods when they are part of a sentence, as in:

$$a + b = \gamma \tag{1}$$

Be sure that the symbols in your equation have been defined before or immediately following the equation. Use “(1)”, not “Eq. (1)” or “equation (1)”, except at the beginning of a sentence: “Equation (1) is . . .”

D. LaTeX-Specific Advice

Please use “soft” (e.g., `\eqref{Eq}`) cross references instead of “hard” references (e.g., `(1)`). That will make it possible to combine sections, add equations, or change the order of figures or citations without having to go through the file line by line.

Please don’t use the `{eqnarray}` equation environment. Use `{align}` or `{IEEEeqnarray}` instead. The `{eqnarray}` environment leaves unsightly spaces around relation symbols.

Please note that the `{subequations}` environment in LaTeX will increment the main equation counter even when there are no equation numbers displayed. If you forget that, you might write an article in which the equation numbers skip from (17) to (20), causing the copy editors to wonder if you’ve discovered a new method of counting.

BibTeX does not work by magic. It doesn’t get the bibliographic data from thin air but from .bib files. If you use BibTeX to produce a bibliography you must send the .bib files.

LaTeX can't read your mind. If you assign the same label to a subsubsection and a table, you might find that Table I has been cross referenced as Table IV-B3.

LaTeX does not have precognitive abilities. If you put a `\label` command before the command that updates the counter it's supposed to be using, the label will pick up the last counter to be cross referenced instead. In particular, a `\label` command should not go before the caption of a figure or a table.

Do not use `\nonumber` inside the `{array}` environment. It will not stop equation numbers inside `{array}` (there won't be any anyway) and it might stop a wanted equation number in the surrounding equation.

E. Some Common Mistakes

- The word "data" is plural, not singular.
- The subscript for the permeability of vacuum μ_0, and other common scientific constants, is zero with subscript formatting, not a lowercase letter "o".
- In American English, commas, semicolons, periods, question and exclamation marks are located within quotation marks only when a complete thought or name is cited, such as a title or full quotation. When quotation marks are used, instead of a bold or italic typeface, to highlight a word or phrase, punctuation should appear outside of the quotation marks. A parenthetical phrase or statement at the end of a sentence is punctuated outside of the closing parenthesis (like this). (A parenthetical sentence is punctuated within the parentheses.)
- A graph within a graph is an "inset", not an "insert". The word alternatively is preferred to the word "alternately" (unless you really mean something that alternates).
- Do not use the word "essentially" to mean "approximately" or "effectively".
- In your paper title, if the words "that uses" can accurately replace the word "using", capitalize the "u"; if not, keep using lower-cased.
- Be aware of the different meanings of the homophones "affect" and "effect", "complement" and "compliment", "discreet" and "discrete", "principal" and "principle".
- Do not confuse "imply" and "infer".
- The prefix "non" is not a word; it should be joined to the word it modifies, usually without a hyphen.
- There is no period after the "et" in the Latin abbreviation "et al.".
- The abbreviation "i.e." means "that is", and the abbreviation "e.g." means "for example".

An excellent style manual for science writers is [**?**].

F. Authors and Affiliations

The class file is designed for, but not limited to, six authors. A minimum of one author is required for all conference articles. Author names should be listed starting from left to right and then moving down to the next line. This is the author sequence that will be used in future citations and by indexing services. Names should not be listed in columns nor group by affiliation. Please keep your affiliations as succinct as possible (for example, do not differentiate among departments of the same organization).

G. Identify the Headings

Headings, or heads, are organizational devices that guide the reader through your paper. There are two types: component heads and text heads.

Component heads identify the different components of your paper and are not topically subordinate to each other. Examples include Acknowledgments and References and, for these, the correct style to use is "Heading 5". Use "figure caption" for your Figure captions, and "table head" for your table title. Run-in heads, such as "Abstract", will require you to apply a style (in this case, italic) in addition to the style provided by the drop down menu to differentiate the head from the text.

Text heads organize the topics on a relational, hierarchical basis. For example, the paper title is the primary text head because all subsequent material relates and elaborates on this one topic. If there are two or more sub-topics, the next level head (uppercase Roman numerals) should be used and, conversely, if there are not at least two sub-topics, then no subheads should be introduced.

H. Figures and Tables

a) Positioning Figures and Tables: Place figures and tables at the top and bottom of columns. Avoid placing them in the middle of columns. Large figures and tables may span across both columns. Figure captions should be below the figures; table heads should appear above the tables. Insert figures and tables after they are cited in the text. Use the abbreviation "Fig. 2", even at the beginning of a sentence.

TABLE I: Table Type Styles

Table Head	**Table Column Head**		
	Table column subhead	***Subhead***	***Subhead***
copy	More table copy[a]		

[a] Sample of a Table footnote.

Fig. 2: Example of a figure caption.

Figure Labels: Use 8 point Times New Roman for Figure labels. Use words rather than symbols or abbreviations when writing Figure axis labels to avoid confusing the reader. As an example, write the quantity "Magnetization", or "Magnetization, M", not just "M". If including units in the label, present them within parentheses. Do not label axes only with units. In the example, write “Magnetization (A/m)” or “Magnetization {A[m(1)]}”, not just “A/m”. Do not label axes with a ratio of quantities and units. For example, write “Temperature (K)”, not “Temperature/K”.

ACKNOWLEDGMENT

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks ...”. Instead, try “R. B. G. thanks...”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

REFERENCES

Please number citations consecutively within brackets . The sentence punctuation follows the bracket . Refer simply to the reference number, as in [**?**]—do not use “Ref. ” or “reference ” except at the beginning of a sentence: “Reference [**?**] was the first ...”

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the abstract or reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors’ names; do not use “et al.”. Papers that have not been published, even if they have been submitted for publication, should be cited as “unpublished” . Papers that have been accepted for publication should be cited as “in press” . Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation .

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.