International Journal of Advanced Technology in Engineering and Science
Vol. No. 12, Issue No. 06, June 2024
www.ijates.com
ijates
ISSN 2348 - 7550

Enriching Cloud Self-healing in Resource management

[1]Saket Suvalal Bhalgat, [2]Dr. P. B. Kumbharkar,

Computer Engineering, JSPM's Rajarshi Shahu College of Engineering, Pune, India

ABSTRACT

To guarantee the optimal performance and availability of cloud-based servers, the process of monitoring, maintaining, and optimizing them is of utmost importance in the paradigm of cloud server administration. This is due to the fact that, in order to facilitate their efficient operation, now a day all applications are deployed and hosted in the cloud. Optimizing for the cloud allows businesses to save costs and increase productivity. Through cloud infrastructure optimization, businesses have the opportunity to reduce storage, computing, and other associated costs. Optimizing for the cloud improves application performance and scalability by optimizing resource use. To optimize these loads, there are a number of approaches, but most of them require human intervention in resource. Accordingly, self-healing and cloud load management are trending subjects at the moment. We evaluate the alternatives keeping in mind that self-healing is essential to autonomous cloud recovery management. To guarantee high availability and reliability inside a decentralized, hierarchical cloud architecture, we present a layered method to cloud self-healing in this work that utilizes the Cosine similarity technique. The results demonstrate that the proposed method achieves good performance with little expenditure when used to medium and small cloud data centers.

***Keywords**: Cloud Resource management, Cloud Load optimization, Event capturing, Event diagnosis, Cosine similarity*

I. INTRODUCTION

Cloud management. Organizational agility, security, and cost-efficiency are all improved with efficient cloud management, which also simplifies administrative responsibilities. The most important reasons to simplify cloud administration for smooth operations are as follows:

1. Streamlining Operations: Simplified cloud management streamlines administrative activities by integrating them into one system, making everyday operations more efficient and less complicated. Among these, we may find automated provisioning, configuration management, and monitoring—all of which work together to cut down on human mistake and boost productivity. To automate the process of setting up cloud environments, for example, with the help of Infrastructure as Code (IaC) technologies such as Terraform or AWS Cloud Formation, setup times can be significantly reduced and consistency guaranteed.

2. Minimizing Expenses: Efficient cloud management helps find and remove wasteful spending by providing insights into resource utilization and the capacity to automate scaling. Companies can better align their cloud

spending with actual usage and eliminate unnecessary expenses by using cost management tools like Google Cloud or AWS Cost Explorer. These tools offer in-depth cost analysis and optimization suggestions. Streamlining these processes allows for the maximization of efficiency and the establishment of clear lines of accountability for expenditure.

3. Stricter Safety and Regulation Compliance: To keep security postures solid, cloud management platforms are essential. These systems should enable centralized control over security policies and compliance monitoring. Simplified management systems provide automated security audits, continuous compliance monitoring, and quick reaction to security concerns. For the sake of risk mitigation and regulatory compliance, this proactive strategy is crucial. Complete security management capabilities are provided by solutions such as AWS Security Hub or Microsoft Azure Security Center.

4. Enhanced Flexibility and Capability to Grow: Businesses are able to increase operations with ease and become more nimble thanks to simplified cloud administration. Kubernetes and other container orchestration technologies, automated deployment pipelines, and continuous integration/continuous deployment (CI/CD) processes allow for faster iterations between development and deployment. As a result, businesses won't have to worry about compromising stability or performance as they adapt to changing market demands.

5. More Efficient Use of Available Resources: Organizations may improve their resource allocation and utilization with the help of simplified cloud management, which gives them deeper visibility into their resource usage. The IT department is able to dynamically optimize resource allocation with the help of advanced analytics and monitoring tools, which reveal performance bottlenecks. By eliminating idle capacity and allocating resources to key applications, overall productivity is enhanced.

6. Collaboration and User Experience: By offering uniform tools and interfaces, simplified cloud management solutions improve collaboration among development, operations, and security teams. Workflow efficiency and problem resolution speed are both improved by this integrated strategy, which eliminates silos and improves communication. This joint effort is taken to the next level with platforms that facilitate multi-cloud and hybrid environments, such as vRealize by VMware or Consul by HashiCorp, which allow unified administration across several cloud service providers.

Achieving smooth operations in today's ever-changing IT settings does certainly need streamlining cloud administration. Simplified cloud management allows organizations to use cloud technologies to their full potential by improving operational efficiency, optimizing costs, ensuring security and compliance, increasing agility, and fostering better resource utilization and collaboration. This allows them to focus on their core business objectives. Organizations that want to stay ahead of the competition and fuel constant innovation must invest in advanced and comprehensive cloud management solutions as cloud adoption keeps rising.

The term "cloud management" describes the process of organizing and overseeing certain aspects of cloud computing. Managed and preserved environments that use public, private, hybrid, or multiple clouds are referred to by this term. It encompasses the processes, strategies, rules, and instruments used for this purpose. Probably right now, we're doing some or all of our work in the cloud. So that we may assess, track, and administer cloud computing assets, systems, and services to our full potential. Resource provisioning and orchestration, automation of cloud consumption and deployment, cost optimization, performance monitoring, security, and resource lifecycle management are all part of the behind-the-scenes effort that keeps our cloud systems working well.

To maximize the efficacy of their cloud installations, businesses employ Cloud Management Platforms (CMPs). To keep their cloud operations and resources running efficiently, firms that wish to use the cloud must manage them. Whether we are working in a public, private, or hybrid cloud, CMP has we covered with a set of tools for managing our cloud computing resources. Companies can save money by using CMP to maximize the efficiency of their resource and service utilization.

The use of CMP has many benefits, including

1. Self-Service Management: Rather of receiving a predefined set of resources, enterprises can take charge of their cloud environment's resources. Using CMP, we may find out if our company is allocating its resources wisely according to what the market demands.
2. Cloud Cost Management: we can save money by having our resources managed by CMP.

Finally, CMPs offer cloud resource management automation in the form of policies and tasks.

The challenges that come with optimizing for the cloud are comparable to those that are associated with cloud computing in general. All of them include: When it comes to deploying to the cloud, selecting the appropriate model Selecting a software as a service supplier that is able to manage our data Acquainting oneself with the potential safety precautions and risks associated with the utilization of an external provisioning service The process of selecting a cloud platform that is both affordable and provides the functionalities that we require We are selecting a system that will prevent our organization from falling. Confirming that the candidate we select has prior experience in cloud optimization through verification

It is tough to optimize cloud computing because there are a variety of challenges to overcome. To begin, an increase in network traffic is a common side effect of increasing the frequency or intensity of cloud computing usage. This is especially true for enterprises that rely on the cloud for tasks that are considered to be mission-critical. It is totally common to behave in this manner. Although many companies have concerns over security, this is a separate issue that needs to be addressed.

Optimizing for the cloud involves a number of challenges due to the fact that it is a relatively new technology. Some of the probable roadblocks include the following: There are some companies who do not provide cloud computing services because the cloud is still in its younger stages. Making sure that all of our data is stored in the cloud is absolutely necessary if we want to make the most of the advantages that cloud optimization offers. The level of security offered by cloud computing is not comparable to that offered by data centres that are physically located on the premises. There is a possibility that the risk of data loss, duplication, and theft is exponentially higher. The construction of security systems is something that needs to begin from beginning. Databases that are managed and maintained in the cloud present a little greater challenge than those that are managed and maintained on-premises. We ought to instead optimize for cloud-based scenarios rather than developing for a system that is located on-premise.

Cloud monitor, discussed in the article by [1] E. Bicici et al., is an initiative to improve work distribution models on cloud computing platforms. The goal is to decrease energy consumption, decrease computation and queue times, and increase resource utilization. provide the author's linear programming framework and any relevant equations for the discussion. First, the author's simulations show that job distributions are better correlated with server energy consumption. Second, there's a chance that resource utilization could be higher in some cases. And third, there would be significant energy savings of almost 42%. The author presents significant findings that can be used to optimize cloud computing platforms in terms of energy efficiency and resource consumption. The development of more efficient and environmentally friendly cloud computing systems is aided by these findings.

Santosh Kumar Sharma et al.[2] proposed an automated diabetes prediction model based on Extreme Learning Machines (ELM) that operates in the cloud. Medical services can reach rural areas and diabetes can be diagnosed sooner thanks to the author's proposed work and cloud computing, which allows continuous services to be offered at any time and from any location. The extreme learning machine is being used in this work because it converges fast, avoids the local minima, and is easier than other traditional classifiers. We have tested the proposed paradigm in both on-premises and cloud-based settings. One way to decrease the feature dimension is by using principal component analysis (PCA). The model gets a better level of accuracy—90.57 percent—by using five distinct features. With the help of cloud computing, the proposed eHealth system can be used as a "Application-as-a-Service." In addition to assisting pathology professionals and doctors, it provides a range of services, including diagnostic detection, which in turn lowers the mortality rate.

[3] Hajar Abedi and et al. describe a cloud-based in-home activity detection and walking period identification system. This system makes use of Internet of Things-based millimeter-wave FMCW radar sensors and sequential deep learning to generate data streams of naturally occurring human activities that take place within the home environment. The suggested system is able to accurately identify the sort of activity that a subject is performing because it makes use of the wealth of continuous data that is available. This system is a significant advancement in the development of autonomous continuous human monitoring systems because it not only detects walking periods and recognizes the type of activity, but it also has the potential to report on the activity level of the subject (for example, sedentary versus active) as well as a variety of other parameters, such as the frequency with which

the subject uses the restroom and the length of time that they sleep. The author produced a data collection of millimeter-wave data obtained from people conducting various activities within their own houses in order to evaluate the performance of the proposed system. This data set also represents a unique and first-of-its-kind resource for the purpose of evaluating the performance of the proposed system.

Due to the extensive integration of many resources, extremely complex cloud environments have arisen in the era of cloud computing. As cloud computing becomes more prevalent among businesses, the efficient and dependable administration of these intricate infrastructures becomes critically important. Because modern cloud environments are getting bigger and more complicated, autonomous systems and services are starting to play a bigger role in them. When faced with novel circumstances, these systems and services may adjust and react on their own. This research delves into the field of autonomous cloud management with a focus on self-healing methods. An crucial aspect for resilient and long-lasting cloud infrastructures is self-healing. Our objective in integrating self-healing concepts into the cloud management architecture is to enhance system availability and reliability.

To account for the complexities of hierarchical, decentralized cloud infrastructures, our suggested method uses a layered structure. Offering a complete answer to the issues of managing large-scale cloud environments, this design encourages reliability while optimizing high availability. To further understand the advantages and consequences of our suggested method, we will go into the details of self-healing processes as they pertain to autonomic cloud management in the sections that follow.

The studies that were considered are examined in the second part of this study. The proposed approach is detailed in Section 3. The experimental evaluation takes place in Section 4, and the study is concluded in Section 5 with a discussion of potential improvements.

II LITERATURE SURVEY

[4] Ranya m. m. salem et.al This research work was conducted by Salem et al. with the primary objective of developing a system that is compact, cost-effective, adaptable, easily configurable, and portable. The system would be able to monitor and control industrial wastewater that is discharged into wastewater treatment plants. Additionally, it would protect workers who are not qualified to deal with such water and would prevent damage to the treatment process and equipment. By evaluating the characteristics of water and the warnings notifications, the system is able to achieve dependability and feasibility in the monitoring processes. This has resulted in the system being more flexible and controllable. This study helps to preserve the natural environment that is comprised of water resources. As a result of the comparative analysis, it was discovered that the proposed system is superior than both the present system and the work that is related to it.

[5] In their discussion of edge computing's many interesting potential uses in industrial IoT and CPS applications, David Hastbacka et.al. Computing that is physically closer to the data source is often required due to the massive amounts of data generated by industrial production applications and the Internet of Things (IoT) devices

themselves. Furthermore, new data-based applications necessitate new models for the construction of secure and efficient application systems. This study laid forth a plan for orchestrating data flows in production environments using data-driven applications, taking into account both the edge and cloud levels. Data flows from the industrial Internet of Things (IoT) were the original target of the method. The constructed model and architecture was evaluated in two separate industrial use cases, with the AHF for SOA infrastructure serving as the foundation. In order to integrate data from IoT devices and production systems, it is possible to dynamically combine cloud and edge application services in a secure and consistent way.

[6] Vingi patrick nzanzu et.al carried out this study effort with the intention of designing an improved resource monitoring architecture for federated cloud infrastructure that was given the term FEDARGOS. The improved monitoring architecture is an extension of DARGOS because of its potentials, which make it one of the best options to satisfy the requirements for monitoring a federated cloud infrastructure that is defined by support for multiple tenants. As part of the process of extending the reference DARGOS, two significant components were modified. These components are the configuration engine, the alarm engine, and, last but not least, an appropriate web-based monitoring console. For the purpose of testing and evaluating its performance, FEDARGOS-V1 was installed on the cloud infrastructure located within the FEDGEN testbed.

[7] According to Muhammad Mu'az Imran et al., it is crucial to closely monitor the manufacturing process and have a good grasp of the elements that can cause deviations in order to ensure that the produced parts meet the necessary requirements. The good news is that additive manufacturing can reveal the topography of each layer because it is layer-by-layer. All of these details are crucial for quality control, including the layer shape and the likelihood of over- or under-deposition. Providing a real-time rasterization method that can detect anomalous agglomerated voxels without additional hardware or post-processing is the objective of this research. The technology is called Defects-Finder. The proposed technique generates structured point cloud data that incorporates spatial and temporal information by rasterizing the incoming data streams in real-time. Contrarily, the rasterization approach's cutting-edge technologies are almost real-time. These technologies, when combined with the need for inter-layer inspection, could greatly lengthen the period that elapses between houses. Defects-Finder, a downstream analysis tool, and this improvement in point cloud processing provide for a significant leap forward in metal additive manufacturing. In theory, it might do away with post-processing altogether and shorten production times.

Junfei Wang et al. [8] propose a contract mechanism to address the pricing and allocation of cloud computing resources in mobile blockchain. One of their suggestions is for the cloud computing service provider (CCSP) to pool user requests for resources so that everyone has the same incentives. Also, to address the issue of knowledge asymmetry, the author proposes an adverse selection-based method. The simulation results show that the CCSP benefits more from the suggested method than from the linear pricing alternative. The CCSP and its users may both benefit from resource pooling in the long run. It is also examined how the solution's performance changes depending on the pool's size and the proportion of various user kinds.

In their thorough review of LiDAR sensors' many uses in critical infrastructure monitoring, Z. Sharifisoraki et al.[9] have laid out all the bases. The author has covered the many uses of LiDAR, such as tracking weather, bridges, ITS, pipelines, mining, nuclear plant safety, and human behavior in relation to environmental protection. LiDAR is also used in smart transportation systems, railroads, and air transportation. Transportation agencies, governments, power plants, and construction companies are among the many organizations that have started integrating these types of sensors into their increasingly streamlined processes in an effort to increase infrastructure safety and decrease maintenance costs. The use of LiDAR has allowed many countries to construct robust and efficient railway networks. Strengthening the train system's resilience is a multi-pronged effort that includes better risk prediction, more thorough monitoring of infrastructure, and preventative maintenance to head off possible infrastructure damages. Trains can quickly scan the rails with the help of LiDAR sensors. In the end, the goal is to have a system of automatic monitoring. Using LiDAR for monitoring lessens the need for human intervention in train surveys, which cuts down on human mistake rates. Another application of LiDAR technology is airport security and automated ground surveillance. When inclement weather is prevalent, air traffic monitoring should be made safer.

[10] Alessandro Tundo a et al. noted that the adaptability of monitoring frameworks and probe technologies for the cloud enables a wide variety of deployment tactics for probes. This may have ramifications for the effectiveness and efficiency of the monitoring system that is ultimately produced. For instance, in order to save resources, numerous probes that serve various operators in a multi-tenant environment can be launched within the same virtual machine. However, this comes at the expense of a decreased degree of privacy and security if the probes are deployed. On the other hand, in order to protect users' privacy, it is possible to deploy one probe for each container or virtual machine. However, this will come at the expense of more resources being dedicated to the monitoring system. In this study, potential techniques for the deployment of probes are methodically derived, presented, and analyzed. As a result, eleven different patterns of probe deployment are defined and evaluated empirically. The outcomes of this work may be of use to engineers in the process of designing their monitoring systems, and they may also generate a set of solutions that can be reused and referred to by individuals. Additionally, the author made all of the experimental material openly available to the public, which included the software artifacts and the dataset that was collected.

In this study, the SHM-IoT MAC4PRO architecture has been completely explained and exhibited for the purpose of assessing the status of two structural targets that are representative of industrial and civil appliances. [11] Lorenzo Gigli et al. In order to achieve this goal, the architecture is designed to abstract from the particular use case and the underlying infrastructure. This is accomplished by integrating cutting-edge SN solutions with the most cutting-edge SHM data collecting, modeling, and processing methodologies. There are four levels that make up the suggested architecture: sensing, interoperability, data management, and service. It is possible to install these layers in an adaptable manner throughout the continuum of the edge to the cloud, taking into consideration the requirements and features of the SHM applications. The adaptability of the author's framework has been proved by the author through two experimental campaigns. During these campaigns, the author provided diagnostic data and addressed the deployment plans for the continuum.

In this article, Deepika Saxena et al.[12] discuss elastic resource management and offer a multi-objective load balancing system based on online predictions. The purpose of the created framework is to maximize resource utilization, decrease power consumption, and minimize the risk of service level agreement (SLA) breach in an oversubscribed cloud environment. Nevertheless, the data center's network traffic was reduced with the implementation of the communication cost aware multi-objective load balancing technique. According to the performance evaluation, the proposed work maximizes resource utilization while reducing performance degradation due to factors such as server overloads, utility consumption, service level agreement violations, data center communication costs, and the number of active servers. All of the findings are backed by the modeling and experiments performed on three separate real workload traces. With the given framework, we may take advantage of the cloud data center's oversubscription environment to significantly increase power savings, even when compared to the most cutting-edge approaches now available.

Job scheduling in heterogeneous cloud computing systems is notoriously challenging, however this work gives an implementation of the Evolution Strategies meet heuristic algorithm to solve this problem. in [13] A group including Petra Loncar worked on this. A component of the offered study was the execution of task scheduling simulations within the CloudSim framework. A single production's ALICE jobs were used to build the simulation's burden. Several simulation experiments have been conducted to establish and validate the performance of the Evolution Strategies approach that has been suggested. The experimental results show that Evolution Strategies' task scheduling method, which uses the Largest Job First broker policy concept, is reliable. Its performance is far better than that of the metaheuristics used by Genetic Algorithms and Evolution Strategies. Assigning work to virtual machines in the most efficient way is within the capabilities of the suggested task scheduling strategy. Optimization of makespan, average execution time, imbalance, resource utilization, throughput, and scalability are all achieved by the Evolution Strategies-based approach. By utilizing dynamic task allocation and managing varied high-capacity data center resources, the system's performance is improved in every circumstance.

[14] According to Amit Sundas et al., the SPMR system is able to constantly monitor patients who are suffering from chronic illnesses that are located in remote areas. These patients include those who have hypertension and diabetes. Both the local and cloud implementations of this framework are equally effective when it comes to forecasting important occurrences like power outages and natural disasters. Through the utilization of a novel CCE optimization approach in conjunction with an innovative DL technique, the authors were able to reduce the disparity between the label error rate that was predicted and the actual label error rate. The possibility that the DL method would achieve efficient convergence is increased depending on the uniqueness of the data. A cloud-based prediction algorithm that is housed on Google Cloud Platform is utilized by the author in order to detect enormous datasets. This work is considered to be groundbreaking. To summarize, the study was able to accomplish its

primary objectives, and the following are the most important findings and contributions: Efficient Remote Monitoring: The SPMR system demonstrated real-time monitoring of individuals who suffer from chronic illnesses, particularly those who are located at a distance, such as people who have hypertension and diabetes. Both local and cloud implementations of the SPMR framework showed to be equally effective at forecasting important occurrences, such as power outages and natural disasters. This was the case regardless of whether the framework was implemented locally or in the cloud.

[15] Qiong Wu et al. offer FedHome, a unique cloud-edge federated learning architecture for tailored in-home health monitoring. FedHome achieves privacy protection by storing user data locally, which is a key feature of the platform. In order to achieve personalized model learning through knowledge transfer, FedHome first collects data from numerous houses, then trains a global model without compromising the privacy of its users, and finally achieves customized model learning. The cloud model that is going to be learned is designed to be a generative convolutional autoencoder (GCAE), which enables the synthesis of samples of minority classes and the formation of a class-balanced dataset during the personalization procedure. This is done in order to address the problem of imbalanced and non-IID data with the goal of overcoming the inherent statistical and communication challenges that are associated with federated learning. In addition, GCAE is comprised of a limited number of model parameters, which enables it to greatly lessen the amount of communication overhead that is required during the process of model transfer. The proposed framework has been shown to be effective in terms of evaluation performance as well as communication efficiency, as evidenced by extensive studies on human activity recognition.

III PROPOSED MODEL METHODOLOGY

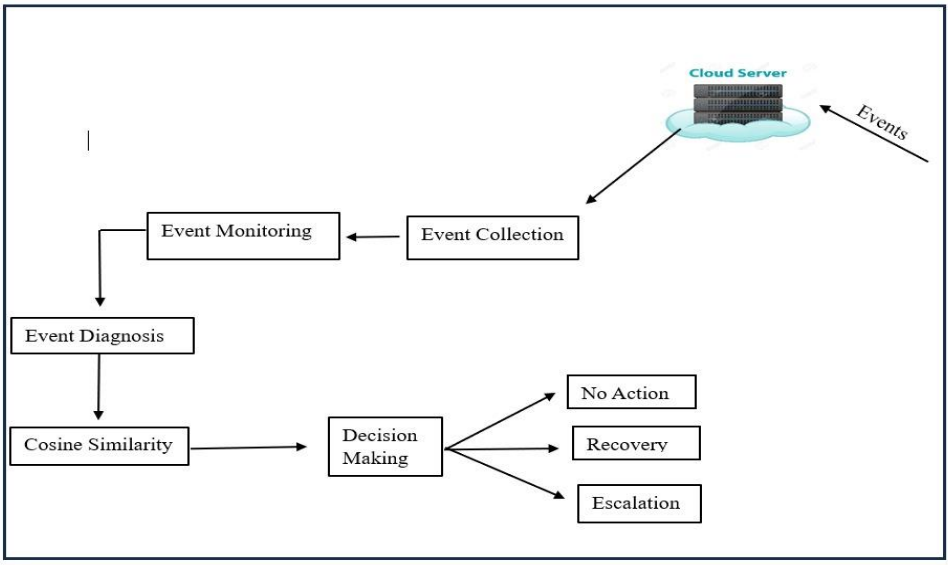


Figure 1: Proposed model for Cloud Self- healing to manage resources

The proposed system for cloud self-healing to manage resources is depicted in the above figure 1. The model utilizes many steps to deploy the above architecture as narrated below.

Step 1: Event Collection - While discussing cloud computing, the term "collection network" describes a system that is specifically built to aggregate and consolidate resources or data from multiple sources into one central point within the cloud. The effective gathering, aggregation, and management of data or resources is made possible by this network's combination of hardware, software, and protocols. A description of its function and application follows:

The following events, listed below, are intended to be collected by the proposed model.

Data/Resource Sources: Data and resource sources might originate from a wide variety of geographically dispersed endpoints, devices, databases, or applications.

Connectivity: The collection network may use application programming interfaces (APIs), protocols (like MQTT for Internet of Things devices), or direct integrations to acquire data.

Data Aggregation: Data aggregation is the process of bringing together and making useable information that has been collected from many sources.

Centralization in the Cloud: Data and resources are centrally stored, managed, and processed on the cloud, which offers scalability, dependability, and accessibility.

Analysis and Utilization: Data and resource centralization paves the way for analysis, visualization, and usage in areas such as business intelligence, machine learning, monitoring, and more.

Step 2: Event Monitoring – During this phase, open-source solutions such as Sensu Core can be utilized. These solutions adhere to an agent-based architecture, which means that agents need to be installed on every host in the system. The agents then do checks on system resources, like disk space usage, and events are monitored by these nodes. In addition, these monitoring packages allow us to build dashboards that show data points gathered from all hosts in an environment. This gives us a better idea of how the Infrastructure-as-a-Code (IaC) architecture is doing overall in terms of health and performance. Proceeding to the next step of event diagnostics requires the obtained list of monitored events.

Step 3: Event Diagnosis - Because the client and cloud provider are both part of the shared responsibility paradigm, diagnosing events in a cloud environment necessitates a different methodology. At this stage, we diagnose the monitored events according to their respective frequencies. Initially, we insert all the monitored events into two lists, List_A and List_B, to estimate their frequencies. From List_B, the duplicate events have been eliminated to have a unique list called List_U.

Next, we estimate the frequency of each event in List_U using List_A. Obainted frequencies are then sorted in descending order to maintain the high priority event to be decisional in the next step.

Step 4: Cosine Similarity and Decision Making – This step involves extracting the event frequency from the double-dimensional list. One column denotes the events, while the other signifies their frequency. This instance vector between the two time windows is subjected to the analysis of the load with the model vector using cosine similarity.

Cosine similarity is a numerical measure of the degree of similarity between two vectors. To be more specific, it compares the vectors' directionality and orientation similarity, ignoring their size and magnitude discrepancies. If the two vectors are to form a scalar when multiplied by each other in the inner product space, then they are required to be components of the same space. Using the cosine of the angle that separates two vectors, we may find their degree of similarity.

The mathematical definition of cosine similarity can be obtained by dividing the magnitudes of the vectors by the dot products of those vectors. As an example, let's have a look at two vectors, A and B. The degree of similarity between them can be estimated by the following equation 1.

$$\text{similarity}(\mathrm{A}, \mathrm{B}) = cos\theta = \frac{A.B}{|A||B|} \;\underline{\qquad}(1)$$

Where

A is the instance vector of load for a given time window

B is the model vector

Dot product of A.B can be provided by the following equation 2

$$\mathrm{A.B} = \sum_{i,j=0}^{n} A[i]\ X\ B[j]\underline{\qquad\qquad\qquad}(2)$$

|A| and |B| represents the magnitude of the vector, which can be given by the equation 3 and 4 respectively.

$|A| =\sqrt{}\ A^2_{[i]}$ ------------(3)

$|B| =\sqrt{}\ B^2_{[i]}$ ------------(4)

The similarity can take on values ranging from minus one to plus one. Larger cosine values are produced by smaller angles between vectors, which indicates that there is a greater degree of cosine similarity. Just one example:

When the orientation of two vectors is identical, the angle that exists between them is equal to zero, and the cosine similarity is equal to one.

Vectors that are perpendicular to one another have a cosine similarity of zero and an angle of ninety degrees between them.

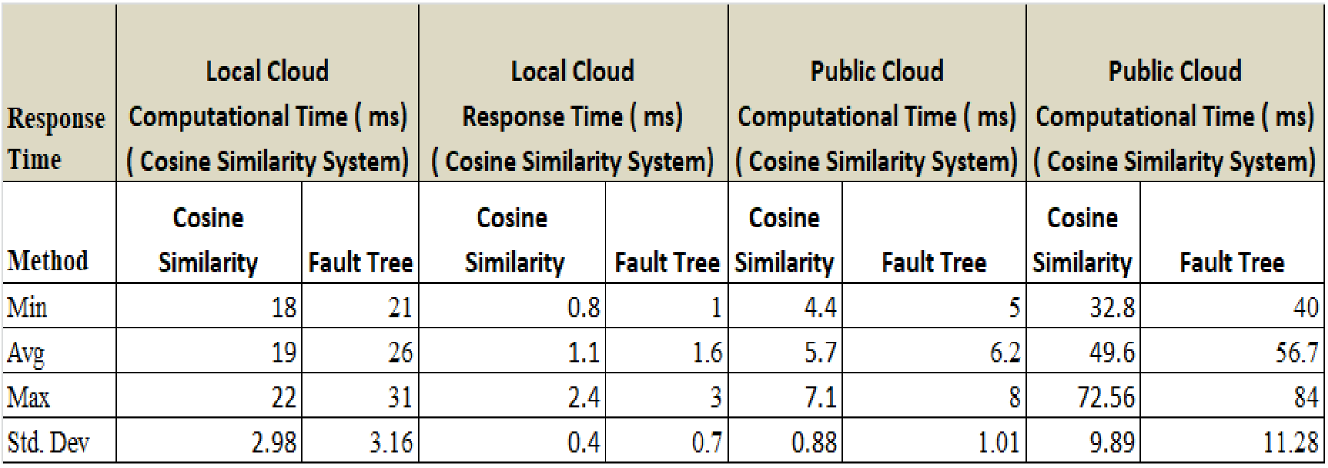
Vectors that are opposite to one another have a cosine similarity of -1 and an angle that is 180 degrees or more between them.

After obtaining the similarity for the specified time window, we make a decision based on the cosine similarity values. This decision includes "no action, ”recovery," or ‘escalation’ to maintain the health of the cloud infrastructure.

IV RESULTS AND DISCUSSIONS

We used the Eclipse IDE to accomplish the described methodology for adaptive self-healing of cloud services by utilizing cosine similarity. Java has been selected as the language for programming and the graphical user interface. The research and implementation of this approach were carried out using a Windows-based laptop with a standard configuration of an Intel Core i5 CPU and 8 GB of RAM.

The method's efficacy and the cloud services' adaptive self-healing capabilities should be tested. A comprehensive review of this kind is necessary, as it must cover every possible outcome that a cloud system can face when implementing the Cosine Similarity. By determining the correct implementation of the cosine similarity for the adaptive self-healing process with respect to the time parameter, we can test the efficacy of the suggested method.



Response Time	Local Cloud Computational Time (ms) (Cosine Similarity System)		Local Cloud Response Time (ms) (Cosine Similarity System)		Public Cloud Computational Time (ms) (Cosine Similarity System)		Public Cloud Computational Time (ms) (Cosine Similarity System)	
Method	Cosine Similarity	Fault Tree	Cosine Similarity	Fault Tree	Cosine Similarity	Fault Tree	Cosine Similarity	Fault Tree
Min	18	21	0.8	1	4.4	5	32.8	40
Avg	19	26	1.1	1.6	5.7	6.2	49.6	56.7
Max	22	31	2.4	3	7.1	8	72.56	84
Std. Dev	2.98	3.16	0.4	0.7	0.88	1.01	9.89	11.28

Table 1: Self- healing response time analysis

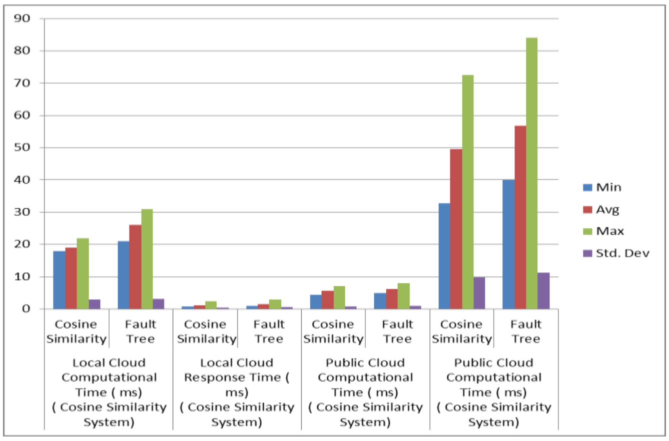


Figure 2: Comparative analysis of Response time in between Cosine similarity v\s FTA

The Response time for local cloud Computational time in milliseconds, local cloud response time, public cloud computational time in milliseconds , public cloud response time in milliseconds for self- healing using Cosine similarity is measured and compared with that of [16]. The obtained results are tabulated and depicted in a graph as shown in the table 1 and figure 2 above.

[16] In order to offer industrial automation systems with real-time supervisory control, a DSS that is cloud-based is suggested. In order to make decisions in real-time, the ASM and knowledge base are moved to the cloud from PLCs. The incorporation of FTA (Fault Tree analysis) methodologies into the cloud-based DSS enables one of the self-management capabilities, self-healing. That the fault tree and the control logic of the PLC may work together without a hitch. As seen in table 1 and figure 2, respectively, our Cloud self-healing system outperforms the model in [16] due to its lack of parallel computing.

V CONCLUSION AND FUTURE SCOPE

This paper propose a cloud-self-healing model that applies the Cosine similarity method to a time window's worth of gathered events. Gathering event data for numerous parameters is done at the outset of the model. The event monitoring method makes use of the Sensu open-source technology. Because both the client and the cloud provider adhere to the shared responsibility paradigm, a distinct methodology is required for accurately diagnosing events in a cloud context. As a result, the observed events are diagnosed correctly. The proposed model categorize the monitored events based on their frequency. A two-dimensional list is generated following the frequency of events examination. We can see the events and their frequency in one column and their names in the other. The load analysis with the model vector is performed on this instance vector between the two time frames using cosine similarity. Cosine Two vectors can be numerically evaluated for their degree of similarity using the cosine similarity metric. More precisely, it disregards differences in size and magnitude and instead examines the vectors' similarity in direction and orientation. A value between minus one and plus one can be assigned to the degree of similarity. A higher degree of cosine similarity is shown by larger cosine values, which are produced by smaller angles between vectors. The cosine similarity values inform our decision-making process once we have obtained the similarity for the given time window. Any choice between "no action," "recovery," or "escalation" will help keep the cloud infrastructure running smoothly. Since the model in [16] does not make use of parallel computing, our Cloud self-healing method performs better when compared to other current systems.

For future deployment, the proposed model can be improved with deep learning models and Transformers that automatically learn new traffic patterns of load in cloud. This improve resource management, and keep cloud load balancing through enriched self-healing approaches. The model can be deployed on real time cloud traffic for self-healing process to keep cloud infrastructure to run smoothly.

REFERENCES

[1] E. Biçici, "A Cloud Monitor to Reduce Energy Consumption With Constrained Optimization of Server Loads," in IEEE Access, vol. 12, pp. 25265-25277, 2024, doi: 10.1109/ACCESS.2024.3365674.

[2] S. K. Sharma et al., "A Diabetes Monitoring System and Health-Medical Service Composition Model in Cloud Environment," in IEEE Access, vol. 11, pp. 32804-32819, 2023, doi: 10.1109/ACCESS.2023.3258549.

[3] H. Abedi, A. Ansariyan, P. P. Morita, A. Wong, J. Boger and G. Shaker, "AI-Powered Noncontact In-Home Gait Monitoring and Activity Recognition System Based on mm-Wave FMCW Radar and Cloud Computing," in IEEE Internet of Things Journal, vol. 10, no. 11, pp. 9465-9481, 1 June1, 2023, doi: 10.1109/JIOT.2023.3235268.

[4] R. M. M. Salem, M. S. Saraya and A. M. T. Ali-Eldin, "An Industrial Cloud-Based IoT System for Real-Time Monitoring and Controlling of Wastewater," in IEEE Access, vol. 10, pp. 6528-6540, 2022, doi: 10.1109/ACCESS.2022.3141977.

[5] D. Hästbacka et al., "Dynamic Edge and Cloud Service Integration for Industrial IoT and Production Monitoring Applications of Industrial Cyber-Physical Systems," in IEEE Transactions on Industrial Informatics, vol. 18, no. 1, pp. 498-508, Jan. 2022, doi: 10.1109/TII.2021.3071509.

[6] V. P. Nzanzu et al., "FEDARGOS-V1: A Monitoring Architecture for Federated Cloud Computing Infrastructures," in IEEE Access, vol. 10, pp. 133557-133573, 2022, doi: 10.1109/ACCESS.2022.3231622.

[7] M. M. Imran et al., "In-Situ Process Monitoring and Defects Detection Based on Geometrical Topography With Streaming Point Cloud Processing in Directed Energy Deposition," in IEEE Access, vol. 11, pp. 131319-131337, 2023, doi: 10.1109/ACCESS.2023.3335130.

[8] J. Wang, J. Li, Z. Gao, Z. Han, C. Qiu and X. Wang, "Resource Management and Pricing for Cloud Computing Based Mobile Blockchain With Pooling," in IEEE Transactions on Cloud Computing, vol. 11, no. 1, pp. 128-138, 1 Jan.-March 2023, doi: 10.1109/TCC.2021.3081580.

[9] Z. Sharifisoraki et al., "Monitoring Critical Infrastructure Using 3D LiDAR Point Clouds," in IEEE Access, vol. 11, pp. 314-336, 2023, doi: 10.1109/ACCESS.2022.3232338.

[10] A. Tundo, M. Mobilio, O. Riganelli and L. Mariani, "Monitoring Probe Deployment Patterns for Cloud-Native Applications: Definition and Empirical Assessment," in IEEE Transactions on Services Computing, doi: 10.1109/TSC.2024.3349648.

[11] L. Gigli et al., "Next Generation Edge-Cloud Continuum Architecture for Structural Health Monitoring," in IEEE Transactions on Industrial Informatics, vol. 20, no. 4, pp. 5874-5887, April 2024, doi: 10.1109/TII.2023.3337391.

[12] D. Saxena, A. K. Singh and R. Buyya, "OP-MLB: An Online VM Prediction-Based Multi-Objective Load Balancing Framework for Resource Management at Cloud Data Center," in IEEE Transactions on Cloud Computing, vol. 10, no. 4, pp. 2804-2816, 1 Oct.-Dec. 2022, doi: 10.1109/TCC.2021.3059096.

[13] P. Loncar and P. Loncar, "Scalable Management of Heterogeneous Cloud Resources Based on Evolution Strategies Algorithm," in IEEE Access, vol. 10, pp. 68778-68791, 2022, doi: 10.1109/ACCESS.2022.3185987.

[14] A. Sundas et al., "Smart Patient Monitoring and Recommendation (SPMR) Using Cloud Analytics and Deep Learning," in IEEE Access, vol. 12, pp. 54238-54255, 2024, doi: 10.1109/ACCESS.2024.3383533.

[15] Q. Wu, X. Chen, Z. Zhou and J. Zhang, "FedHome: Cloud-Edge Based Personalized Federated Learning for In-Home Health Monitoring," in IEEE Transactions on Mobile Computing, vol. 21, no. 8, pp. 2818-2832, 1 Aug. 2022, doi: 10.1109/TMC.2020.3045266.