International Journal of Safety and Security Engineering
Vol. 12, No. 6, December, 2022, pp. 681-690
Journal homepage: http://iieta.org/journals/ijsse

Mapping of Virtual Machines Using Machine Learning Algorithms for Detection of Faulty Nodes

Reshma S. Gaykar[1*], Velu Khanaa[1], Shashank D. Joshi[2]

[1] Bharath Institute of Higher Education and Research, Chennai 600073, India
[2] College of Engineering, Bharati Vidyapeeth, Pune 411043, India

Corresponding Author Email: reshma.gaykar@gmail.com

https://doi.org/10.18280/ijsse.120603

Received: 29 June 2022
Accepted: 17 December 2022

Keywords:
fault detection, distributed system, dynamic job ordering, virtual machines, machine learning

ABSTRACT

A distributed system is characterized by a large number of nodes that are linked to a network and are mostly used for transaction processing. Large set of users are likely to communicate information over the network to the nodes, consistency and dependability remain a critical problem in the distributed environments. Independent failure of the component is one of the major problems in the distributed systems as it slowly impacts the performance of the other nodes in the system. The quality of service - QoS of a distributed network may be improved by a quick way of detecting problematic nodes. Sometime heavy nodes required high computation for transaction processing while idle nodes take low computation. In this paper, we proposed identification of straggler nodes in distributed environment with the help of hybrid machine learning algorithm. The work basically carried out to set up of large number of virtual machines and collect current log audits of each VM. According to the available parameters of audit files to each machine, algorithms decide that specific node is overheated or ideal condition. In expensive experimental analysis we demonstrate a accuracy of proposed hybrid machine learning algorithm. The proposed algorithm produces higher precision up to 4.5% than state-of-art methods. Key highlights of the VM mapping strategy were also investigated through a scrutiny of ongoing contracts. Main focus remains on machine learning (ML) to distinguish PM (Physical Machine) congestion, determining VMs from crowded PMs, and VM conditions as major exercises. This paper aims to review and characterize research on the planning and status of VMs that use ML using asset usage history. Energy productivity, VM migration, and quality of service were the main exhibition boundaries used to investigate cloud data center presentations.

1. INTRODUCTION

In distributed processing architecture technique, perception divides statistics into fragments and distributes them to all slaves, from which it receives a large amount of data and replicates it to slaves, which could then be processed simultaneously on heterogeneous groups to complete the task faster. The situation known as stragglers is a natural result of this sort of parallel framework proposed, i.e., gradual operating nodes, which likely postpone the overall activity final touch. Straggler tasks continue to be a major impediment to attaining a faster finishing touch of information of parallel and computation-intensive applications while working on trying to cut humongous, amended section in the cloud. We need straggler-tolerant strategies for this, which aid in the sequencing of a set of rules for reducing the effect of stragglers. It's critical to recognize the factors that cause a node to be a prone to failures node when recognizing straggler vertices.

The similar approach is resource matchmaking to assign right job to right VM, it is very essential process to complete respective job execution in desired time. When input sources generate random jobs, it categorized into a small, medium and large types. When job size is large, it is necessary to assign those jobs to large virtual machines for hassle free execution. In above Figure 1 we demonstrate dynamic job ordering for classification of jobs while workflow scheduling has used for checking all resources such as connected VM's. After calculating the nodes parameters such as CPU load, memory load, number of task execution etc., algorithm predicts the possibility of perfect resource for respective job by using proposed hybrid machine learning classification. The approach can reduce the computation time in large distributed environment even in heterogenous data processing resources. Furthermore, the remainder of the work is separated into the following sections: Section 2 discusses different current approaches for defective node identification in distributed frameworks developed by earlier researchers. Section 3 describes techniques used in the prospective system's design in details, whereas Section 4 illustrates the algorithm used for implementing the proposed system. The experimental process for examining the findings gained with our proposed technique, as well as comparison of analysis with several state-of-the-art methodologies, is described in Section 5. The proposed system is concluded in Section 6 along with and its future work.

A data center consists of a small number of PMs or servers with many CPUs, memory, disks, and bandwidth (BW) assets. Proper use of these resources is an important task for merchants to keep up with quality of service and energy usage costs. Keeping up with the nature of management and energy costs is an important challenge when considering distributed

computing. To achieve this, we are effectively conducting various surveys in the field of data center asset utilization, with VM reservations and status as the basic issues. Cloud specialists need to deploy upgraded VMs in the data center while addressing SLAs and power usage costs. This is a NP-difficult issue and can be settled utilizing a container pressing methodology. A great deal of techniques was proposed utilizing heuristic, meta-heuristic and ML-based answers for streamlining. In the vast majority of the ML techniques, verifiable information or responsibility follows were utilized as a necessary contribution to foresee future asset usage. The expectation of asset use requires framework conduct by advancing subsequently different calculations were created to make a precise expectation for VM the executives. VMs the board is applied when the necessary assets are being utilized totally. Yet, on account of less asset use PMs experience the under-stacking circumstance. Additionally, because of static distribution, these assets can`t be reallocated among PMs. The answer for such an issue is dynamic VMs solidification that can be created utilizing the different ML procedures. In unique VM combination, the choice of VMs was finished for relocation in reasonable PM to keep away from PM over-burdening and under-stacking. The majority of the examination work was finished for dynamic asset the board utilizing the ML methodologies on current and past asset utilization information. This information just gives data about usage values in a specific span of time. It gives no data about asset usage later on, so different methodologies are being created to anticipate the use of assets.

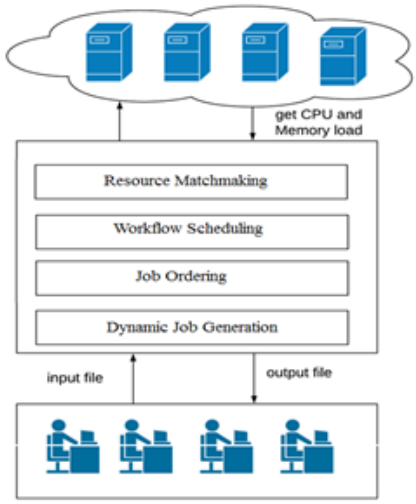

Figure 1. Resource matchmaking scheme in distributed environment using machine learning techniques

2. RELATED WORK

According to Sakir et al. [1] proposed system enhanced faulty node identification for dispersed systems using the C programming language weighting factors. For dispersed networked manipulating systems, a better faulty node detection strategy using the C programming language weighting aspects, which video display unit's node conduct using pseudo-random Bose-Chaudhuri-Hocquenghem (BCH) code. Every slave node's single-bit BCH code is used by the master node to gather the alternative of the cyclic redundancy check (CRC) codes. However, without the attention of channel nodes, BCH code may easily gain the suspected faulty node's errors function. As a result, suspected errors nodes are recorded in the detected mistakes c programming language and normalized for interpretation using the weighting thing, which is conducted at some point in the sequential check. Fault judgement is used to completely evaluate the data and ensure that the reported errors are accurately recognized. The data is represented using statistical features of raw and filtered records.

In WSN [2] a brought together Faulty Node Detection rule helped applied math examination. In a remote gadget organization, a concentrated damaged hub recognition framework in light of measurable examination. The proposed method is tried, and the recreation results uncover that it beats the ongoing traditional strategies. Shortcoming Detection in Wireless Device Networks: A Performance Analysis of Various Algorithms [3] the different past techniques and introduced a more thorough meaning of issue 1detection and 1fault resilience in WSNs. This study gives an outline of current imperfection location draws near, as well as additional examinations to help sensor applications. Over Device Network Distributed Intermittent Detection Accuracy for Linear Randomized Systems [4]. In sensor organizations, a remarkable extra generator is intended to see the value in the scattered discovery of Influence Factors (Ifs). Weight network of the outstanding worth acted in this article are administered by a lack of quality file of the former gauge to decrease the spreading impact of IFs, as opposed to standard moving skyline assessment methods. Number cruncher boundaries are made utilizing the network change approach and factual and numerical hypothesis, and the awareness of one IF is surveyed utilizing the remaining. To forestall recognition results from independent residuals impacting, the world perceptibility necessity is applied to all IFs.

Disseminated Fault Detection and Isolation for Second-Order Multiagent Systems are used with Exogenous Disturbances Using Partial Nodes [5]. A strong dispersed shortcoming recognition and disconnection (FDI) topic for second-request multiagent frameworks with outside shocks will be given, which will uphold fractional specialists' outright state. This plan makes areas of strength for a standard with an unsettling influence dismissal term for every specialist, trailed by a further developed MAS control framework with no aggravation term. Making a time span for Node Failure Prediction is accessible with Aarohi [6]. Aarohi is a structure that offers a successful method for estimating disappointments on the web. Since Aarohi is general and versatile, it very well might be utilized as a constant indicator. For a series length of 18, Aarohi accomplishes three-minute lead times to hub disappointments, with a middle expectation season of 0.31 milliseconds. A stateless hub disappointment ID procedure that recognizes a hub disappointment at a similar speed as the ongoing plan without keeping up with the proliferation target associations [7]. Each machine hub in the proposed procedure will successfully involve machine assets for its specific work. At the point when a hub disappointment is recognized then this approach computes the engendering focuses as opposed to holding the spread targets. This approach gauges the accuracy of a basic model that guarantees powerful spread. The overlay distance between the propagator hub and bombed hub was found which is critical to decide the precision. The framework modifies the keep alive span to predisposition the disclosure of adjacent disappointment hubs, which upholds this finding [7].

Slacker Node Identification in Distributed Atmosphere Victimization Algorithms is utilized for Soft Computing [8], a framework for distinguishing stray hubs in a disseminated climate utilizing an AI procedure for a huge scope task executed log information. At the point when an AI calculation is run, the framework will expect the loafer hubs and powerfully eliminate them from the execution list. Q-Learning is a support learning technique anticipated for execution and approval in multi-hub frameworks all along [8].

Faulty Node Detection in a Highly Distributed Environment Victimization [9] A Hybrid Machine Learning Approach In distributed settings, the identification of malfunctioning nodes. Since some nodes are already hot, the main goal of this effort is to see how the network handles the massive amount of data. Once the master node provides a replacement task to such nodes, the time complexity measure is linear unit rate. Such issues might have an influence on service quality as well as service decline. To resolve such challenges, first do an examination of available resources, and then choose the best ode. In distributed virtual machine settings, a hybrid deep learning technique for detecting problematic nodes has been developed. Since then, gather a variety of logs from each virtual machine and extract a variety of alternatives such as variable features, relative features, bi-gram features, and so on.

To safeguard against different hub disappointments, affiliation data handling utilizes a quick reroute component [10] is the main concentrate in the IP quick reroute examination space to manage numerous hub disappointments. In the organization style stage, the proposed strategy creates crossing trees to sidestep disappointments from a given geography, and in the organization activity stage, it reroutes a parcel by means of one of the delivered traversing trees each time the bundle hits a hub disappointment. This recommended approach may effectively make such spreading over trees, as shown by a mathematical model. To further develop Node Fault Tolerance Capability, utilize the Failure Node Reduction equation [11]. The addresses various strategies to give able approaches to upgrading Cloud show in its entire, giving the client a considerably more OK and capable climate. Calculations and techniques for load evening out are depicted. Every technique was made in light of a particular objective, for example, versatility, elite execution, legitimate asset use, etc. Notwithstanding, there are sure snags in the heap adjusting strategy, for example, above, that should be survived. To diminish the quantity of disappointment hubs, the base distance between assets will be figured, and traffic will be coordinated to the assets with the briefest separation from their neighbours [12, 13].

To improve QoS in wireless device networks (WSNs), a cost-effective protocol for node failure detection has been developed [14]. The matter of detection is cut by the wireless sensor network's remaining nodes. This paper proposes a formula which allows each node to determine whether the property to a particularly chosen node has been lost. It also determines the prevalence of the cut for one or more nodes (connected to the special node at the time of the cut). The method is asynchronous and distributed in nature, which means that each node may only connect with nodes that are int the range of communication. The technique is based on the calculation of a false "electrical potential" of the nodes over and over again. The basic iterative subject's assembly rate is free of the organization's size and geography. Start to finish looking through procedures empower savvy revelation and recuperation of hub disappointments [15]. Since network support is a particularly significant part of systems administration, the work has tended to an assortment of deterrents, and it has likewise taken care of an assortment of issues as far as organization execution through legitimate checking. The fundamental objective of this venture is to find the issue hub and its area utilizing different organization testing procedures. This has been achieved by utilizing a few strategies, for example, twofold drawback to find the disappointment hub and by utilizing an alternate arrangement of rules for intelligently exploring the ways and finding and convalescing the disappointment hubs. Bangare et al. [16-20] have contributed Machine learning projects for clinical pictures. Shelke et al. [21] and Gupta et al. [22] have some notable work in research field with LRA-DNN and emotion extraction. Scheduling and resource management strategies for the cloud are referred from the references [23-27]. Pande et al. [28-30] worked for the spline methods etc. Used the basic concept of straggler and ML [31-34].

The above literature describes work done by previous researchers but still the used design, frameworks and algorithm having some gaps such as high error rate, module overfitting problem when it deals with supervised classifiers. Also, most of the papers has insufficient knowledge of data extraction from audit data. Some of the papers are tested on the homogenous configuration which tends to give the good rate of success but fails when tested on the heterogenous configurations.

3. PROPOSED SYSTEM ARCHITECTURE

The proposed research describes the design and develops a hybrid machine learning algorithm for faulty node detection from the distributed environment. The various feature selection methods have been used to classify faulty nodes. The CPU and Memory utilization are the resource parameters considered in the predictive algorithm. The usage or load of a VM indicates how much of its capabilities are being used by the VMs that reside on it. The terms of a computing power are source of use. On just one extreme, it is in the CP's best interests to maximize CPU usage to maximize recourses' use. On the other side, if the CPU load is too high, it's more likely that the VMs on a particular VM won't get the capacity they need, resulting in SLA breaches and lowered service quality. Computational resources load might cause electrical and hasten the ageing of the hardware. Many academics focused on CPU load as a result of these factors. Other resources, such as memory or disc space, might, nevertheless, create a bottleneck. The cache is of particular significance since present virtualization approaches do not assure that the cache utilization of various VMs supported by the same PM is isolated, resulting in contention. As a result, it's critical to understand and forecast the information that might occur when two virtual machines are co-located. The energy usage of a VM increases in lockstep with the CPU load. Knowing the precise relationship between power requirements and CPU load is a difficult task in and of itself, and even its implementation. The load on some other components (such as the disc) may also influence. However, over several use cases and architectures, linear approximations of power requirements as a function of CPU utilization work pretty well. The entire execution produces as an outcome of faster performance and eliminates the possibility of data loss during the execution.

As shown in Figure 2 the jobs are submitted to the master node then those are executed on the different VMs. We collect the memory and CPU load on the different VMS for the further processing.

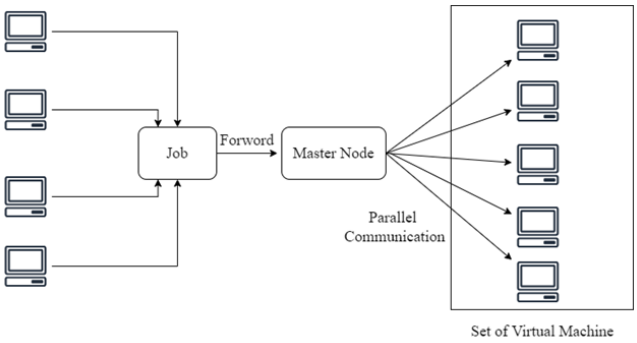

Figure 2. Job submission mechanism

Fluffy far-reaching assessment is placed ahead in light of the nonlinear qualities of the assessment cycle; it is the utilization of the fluffy activity rules in fluffy math, the assessment of nonlinear space to measure the union cycle to be the quantitative assessment results similar. Lately, document-based work process has been increasingly more broadly utilized in the film and TV creation industry [10]. The entertainment world necessities to supplant the film and tape sooner than the broadcast business, and the media business is an option to the tape [11]. Computerized middle of the road innovation in the film making industry is certainly not another idea, it has turned into the standard creation cycle of world film post. One of the significant highlights is that the autonomous edge document is utilized as the visual data transporter and the essential handling unit [3]. Presently the Internet depends on client/server-based application model design, the application should set up a server in the organization, can move data through the server [12]. Data or first transfer saved to the server, then, at that point, download, or data on the server as indicated by the restrictive principles after treatment can be disregarded the organization flow [13].

However, in the event that you utilize the product sharing framework, conventional PC can't straightforwardly through the server and one more with a similar portion of the PC framework programming structure the Internet, sharing assets, joint effort to finish some action [16]. All what's more, similar shared programming hardware and clients, can frame a for its own common confidential organization on the web.

The client / server model which focus on the Internet, regardless of information resources or cost resources to focus on the same direction, this pattern is consistent with the one to many and strong to the weak form of social relations, such as the government on individuals, enterprises, large enterprises to small businesses, schools, enterprises of workers etc. [17]. And sharing will lead to the amount of information, cost resources are evenly distributed to each point of the Internet, which is the so-called "marginalized" trend. This model is consistent with the characteristics of "one to one", as well as the forms of social relations, such as individuals to individuals, enterprises of equal size, etc... Therefore, there are two ways to coexist and complement each other. In fact, sharing technology as a new computing mode and new network applications, reflects the forms of organization of human society: everyone to contribute their resources, enjoy the resources provided by others; is the collective human behavior in the current technology, the natural refractive index in the field of information technology, will exist for a long time, and with the further improve computer processing capacity and network capacity and the evolution and development.

Information content storage and exchange is one of the most successful applications of shared technology. The implementation of shared file sharing system focuses on the following issues: Information positioning: positioning information in the shared file sharing system plays a very important role, because it gives the users of the system provides the basic information sharing resources, such as which nodes and resources which are available in the network, so that nodes can be directly linked to other nodes about the resources required for system [18]. File transfer: file transfer refers to how to transfer, distribute and copy shared files between nodes in a shared file sharing system. Figure 3 show how to read and process the file steams in a network. In Internet, the main transmission modes are single to single transmission, multiple to single transmission, multi to multiple transmission. For example, a system will split large files into multiple files, the node can simultaneously from multiple nodes to download the file blocks, while their file blocks available to different nodes, to achieve multiple transmission. Coordination and collaboration with excitation: refers to the shared file sharing to establish cooperation relationship between the peers in the system to complete the task, without going through the central server to collect and broadcast information, so as to realize the sharing of resources. However, many rational users are always trying to use other people's resources more and less to contribute their resources. How to motivate users to contribute more to their own resources and ensure fairness in the exchange becomes a key issue in shared file sharing system.

In recent years, the shared file sharing system has been very popular in Internet, this is mainly because it can successfully and efficiently locate and copy the contents of file information distribution. Information positioning has attracted a lot of attention in the past few years, and a great deal of research has been made. At present, the distribution and replication of files in the shared system has become a very active research topic recently. Research on the distribution and replication of information file in shared file sharing system is going on. Figure 4 show different components of the stream and its hierarchy.

Customers request assets on demand and run their applications with VM assets that match their application needs. In most cloud frameworks, executive VMs ended up with continuous resource usage. Interactions between VMs and executives require additional computational and memory resources that increase frame costs and affect overall execution. In our review, the behavior of the learning framework has emerged as a good strategy for nurturing sophisticated VMs in the executive cycle. This may be suggested by past, generally long-range usage levels. It is an essential test to designate VMs onto PMs, while keeping up with the service level agreement (SLA) prerequisite, framework execution, ideal usage of assets, and decrease in energy consumption (EC). Over-burdening and under-stacking of PMs is additionally the issue that happened in the arrangement cycle, to stay away from such issues the forecast of asset use was finished in the majority of the exploration. The expectation of asset use isn`t just in view of current utilization designs yet in addition past framework conduct might be thought of. In this audit, it was found that AI (ML)

models are appropriate to foresee asset usage utilizing verifiable information to accomplish viable VM planning and situation. This paper audits various areas of writing that arrangement with the Virtual Machines (VMs) position onto physical machines (PMs). This overview aims to use an ML approach with asset usage history to distinguish and group VM bookings and surveys in the area of location.

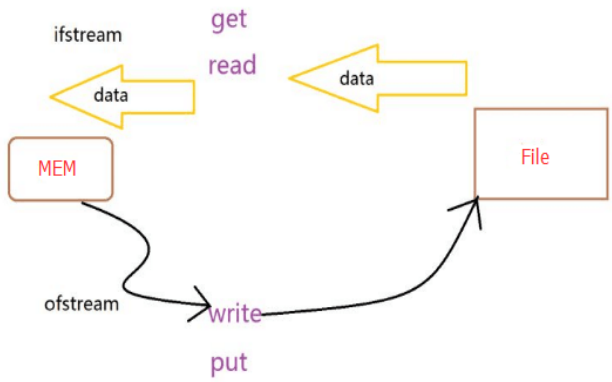

Figure 3. Read and write process of file stream in a network

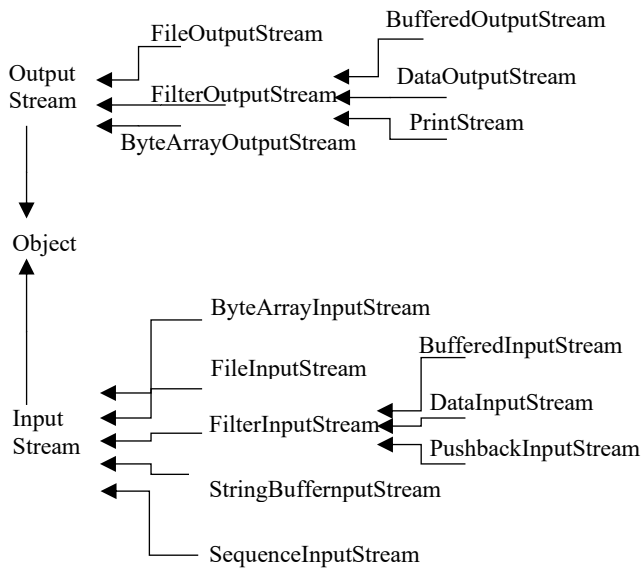

Figure 4. File object flow

3.1 Resource utilization of virtual machine

Resource usage and garage had been applied to preserve facts in a file (utilization follows) that consists of the usage of belongings in a right time stretch. Usage follows are anticipated to assume the framework behavior with the aim that destiny use of belongings may be assessed for powerful VM making plans and role withinside the cloud datacenter. Handling, memory, plate, switch pace and I/O use statistics are to be saved up with in a file with the aid of using use display so the cloud framework can surely observe the use measurements of VMs and PMs for expectation. Asset utilization of VMs is essentially involved approximately the use charge in a term and PMs Resource use measurements are taken into consideration as whole use of VMs (facilitated with the aid of using the evaluating PM). The professional primarily based totally method become applied to shop VM approaches of behaving (asset use over the lengthy haul) in a international facts set to make open for all specialists [14]. RL (reinforcement Learning) primarily based totally studying professional observed the PM fame and gathers the continuing absolute utilization of PM [19]. A technique become brought to display and amassed PM asset use using the close by professional and the global professional one at a time to foster relocation components [20]. The checking of belongings become completed to accumulate asset use facts and amassed on diverse instances of every second, 1, 10, and 30 min of belongings [21]. A checking motor become completed to accumulate asset use facts from VMs. It became applied to accumulate asset usage facts in a bit time body for putting in place that facts in a record. The facts contained CPU, memory, circle and IO measurements almost about usage.

3.2 Mapping of VMs

When it detects an overcommitted or under stacked PM, it starts the VM relocation history and continues to adjust the PM heap. All VMs from the underutilized PM will be moved to the normal PM and the underutilized PM will be powered off. Due to PM congestion, some VMs for moving PM must be congested to achieve ideal load matching. Determining which VM to relocate the overloaded PM is an important task. When the VM is moved to the PM, it is important to reconfigure the VM for the new PM to improve execution. The VM selection scheme has been applied to moving VMs, reducing post-movement power usage compared to other VMs assigned to similar PMs [22].

4. ALGORITHM

The proposed Machine Learning (ML) algorithm describes the overcome of Virtual Machines load prediction in distributed environment. It illustrates the overall machine data parameter and its values calculated by different feature extraction techniques. The following algorithm is used to generate the virtual machine's load as per the sum rule prediction approach.

In: Parameter values in map<integer value, string class> having attribute values for Virtual Machine. Policy patterns {pp1, pp2, … ppn}
Out: Individual report for Virtual Machines

1: for loop – for every read of a map

$$ext\_attrb[k][l] \sum_{k=0,l=0}^{z} \left(a_{[k]}, a_{[l]\ldots\ldots\ldots}a_{[z]}, a_{[z].}\right)$$

2: if ext_attrb [k] similar to pp[1]
nrmlpos = +1
nrmlmasterlits. add ← (nrmlpos)

3: if ext_attrb [k] similar to pp[2]
abnrmlpos = +1
abnrmlmasterlits. add ← (abnrmlpos)

4: if ext_attrb [k] similar to pp[n]
deniedpos = +1
denmasterlits. add ← (deniedpos)

5: for loop end.

6: for all the above class lists calculate the fitness factor using following formula.

$$f = \sum_{k=0}^{n} \frac{F(x)}{SumF(x)}$$

7: currntlst_weight [w] = $\frac{master\_list[i]}{totalvm}$ *100

8: Sort the current list currntlst_weight[w] in descending order

9: Put the first element of currntlst_weight list variable into a recmd_currentlist for final class for VM profile.

10: end process

The above algorithm describes the detection of anomaly nodes using statical analytical features. We extract all features from the local repository with all possible attributes in the first step. The second step removes all features from attributes and validates them with defined policies. According to procedures, it decides an event is expected, abnormal or dangerous. In the last phase, we calculate the mean values, generate each list's score, and optimize using descending order. The zeroth position of the list the generated event as an outcome of the algorithm. At present, many shared files are large files, such as most of the multimedia file, then the system nodes transmit data by what way, how will these large files quickly distributed to each node in the system, whether the system can well support the massive download node while downloading file sharing is a careful study must be shared file sharing system problems [19].

4.1 Management of VM based on ML

Much of the research has developed various techniques for predicting CPU utilization using the following responsibilities: Various ML models can be used to facilitate VM booking methods for matching the load on PM assets as shown in Figure 5. The ML model provides the cloud framework with the ability to consistently advance and evolve the execution of the framework used to reserve and deploy VMs in cloud server farms. It also uses verifiable information to facilitate components for learning and predicting future asset use. The most common type of planning begins with the recognition of genuine information. Use of previous CPU, RAM, and disk to find examples of information to improve expectations for future use. In a cloud framework environment, huge amounts of information are processed with high computational requirements. Therefore, providing updated EC, cost, and SLA compliance to executives is a fundamental issue in managing VMs. ML can be used to evolve VM boards in cloud data centers to manage vast amounts of information and huge asset assumptions. The ML model allows you to make better predictions to enable strong asset management and full EC in your data center. Accountability information helps the ML model learn more about the framework's behavior and asset usage expectations. Responsibility prediction helps facilitate a working cloud framework. Our review found that CPU utilization expectations are the most moving area of research in distributed computing to improve VM board cycle times while catching up with SLAs and ECs. Our review found that since 2009, the ML model has been applied to verifiable information, facilitating improvements in the components for planning VMs and PMs over time. Therefore, using historical asset usage information is a widely accepted approach in cloud data center VM planning and status research environments. ML-based computations have been used in writing to facilitate VM asset sharing, scaling, and live relocation in cloud frameworks.

4.2 Support Vector Machine (SVM)

SVMs are used to provide answers to multifaceted display problems by using the idea of selection hyperplanes to characterize selection boundaries that separate the placement of objects of different classes. This is a regulated ML model used to characterize information and recurrence [23]. SVMs have been used to facilitate dynamic and versatile VM reservation calculations with respect to asset usage tips from VMs and PMs. The asset analyst used the PM and VM usage history to place the information index in a common area. This dataset was used to predict future asset usage for VM reservation and situation techniques. The progress of the VM planning was completed by characterizing the PM load according to the asset usage of the VM using the SVM. Authentic information of PMs and VMs were assessed and arranged by the general asset use [24]. In some exploration support vector relapse (SVR) was likewise utilized as a learning approach and SVM group information and make expectations successfully on cross-over locales of two classes, it is generally utilized.

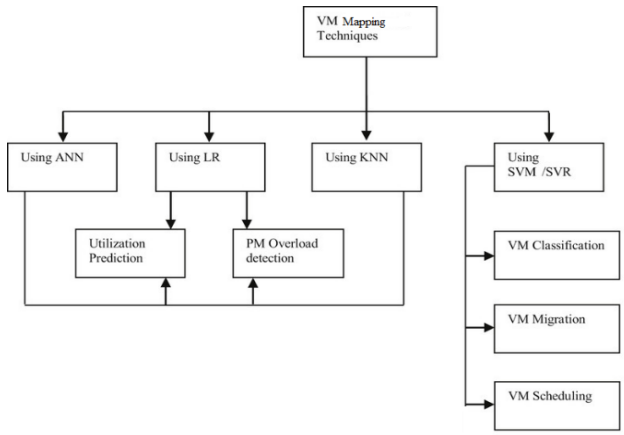

Figure 5. Virtual Machine Mapping techniques

4.3 Artificial Neural Network (ANN)

ANN has an info layer, hidden layers, and a result layer. Each layer has specific quantities of hubs (neurons), coordinated with the sigmoid enactment work. This method gives self-learning components in view of the idea of neurons, associations and move capacities. Research works in light of brain networks show that the expectation of asset usage is a powerful way to deal with deal with the assets in the cloud datacenter. The brain network is utilized as a productive apparatus for determining in different sorts of exploration issues. It is also applied to infer the need for resources in distributed computing. It is typically used to remove and analyze the actual instance of responsibility and predict future responsibilities for the data center in the near future [25]. The verifiable or accountability sequence is divided into expected model preparation and test information. In much of the

research, the demand for CPU resources at each point in the verifiable information was commonly used as information for predicting resource usage. Furthermore, it was thought to contribute to the spread of brain tissue throughout the tissue with a certain weight. The result layer was utilized to give a result signal after the handling of neurons. Subsequently, the info signal was considered as the CPU prerequisite at a specific time. Computer chip prerequisite for whenever span was addressed by the result signal. Brain networks are better ML ways to deal with foresee CPU usage. The forecast of future CPU, Memory, Disk, organization or other required assets in view of authentic information will distinguish unused, under-stacked and over-burden PMs [26, 27]. This information helps you determine which PMs should ideally be created and which PMs should be moved to reduce data center energy consumption and costs.

5. RESULT DISCUSSION

The system is tested on four different nodes and a single dedicated node characterized by an open regional distribution network. The system under test was loaded continuously for each data node using a randomized task creation and techniques utilized ordering approach. All he data nodes under the testing environment has the same configuration applied so that the resulting outcome can be compared against each other's. Each node's different properties, such as IO load, CPU load, and data proximity features are read and trained using a supervised classifier model. The system is compared against an independent learning method, as shown below in Figure 6.

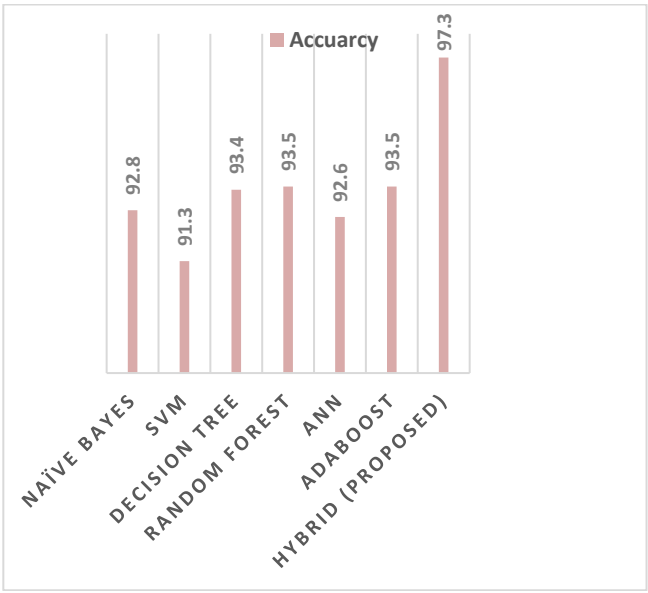

Figure 6. Faulty node detection accuracy with various traditional machine learning algorithms as well as proposed hybrid machine learning algorithms

At the very least, an estimate of the techniques' cumulative harshest runtime and memory usage should be provided. An assessment of the methods' exponential average-case behavior would be even more fascinating if technically achievable. An investigation of certain more straightforward instances, on the other hand, might help to a deeper comprehension of and hence enhance confidence in the suggested algorithms.

5.1 NB TREE classifier

Naive Bayes tree is a class for producing a choice tree with guileless Bayes classifiers at the leaves. A choice tree is a choice help instrument that utilizes a tree-like chart or model of choices and their potential results, including chance occasion results, asset expenses, and utility. Showing an algorithm is one way. Choice Tree is a stream outline like design in which inward hub addresses test on a characteristic, each branch addresses result of test and each leaf hub addresses class mark (choice taken subsequent to processing all credits). A way from root to leaf addresses order rules.

A choice tree comprises of three sorts of hubs:

1. Decision hubs regularly addressed by squares.
2. Chance hubs addressed by circles.
3. End hubs addressed by triangles.

5.2 Hybrid method using ML algorithm

Following pseudo code is used to produce the result of a Hybrid method described in the paper.

dom(makeNBTree),
Set LabeledInstance
cod(makeNBTree),
Tree Split NBC

a. For each characteristic Xi, assess the utility, u(Xi), of a split on quality Xi. For constant credits, an edge is likewise found at this stage.
b. Let j = argmaxi (ui), i.e., the property with the most noteworthy utility.
c. If uj isn't fundamentally better compared to the utility of the ongoing hub, make a Naive-Bayes classifier for the ongoing hub and return.
d. Partition the arrangement of examples T as per the test on Xj. In the event that Xj is constant, an edge split is utilized; assuming that Xj is discrete, a multi-way split is made for all potential qualities.
e. For every child, call the calculation recursively on the part of T that matches the test prompting the youngster.

- Utility of Node: Computed by discretizing the information and registering 5-overlay cross-approval precision gauge of utilizing NBC at hub.
- Utility of Split: Computed by weighted amount of utility of hubs, where weight given to hub is corresponding to num of occasions that arrive at hub.
- Importance: Split is critical if the general decrease in mistake is more noteworthy than 5% and there are no less than 30 occasions in hub.

The outcomes acquired uses NB tree approach is displayed in Table 1.

As per the experimental outcomes displayed, the crossover model that gives a superior generally execution is likewise greater at recognizing irregularity assaults than the customary and expanded Naïve Bayes strategies and the KDD'99 champ. We likewise analysed the best outcomes acquired in our examinations with the tantamount outcomes from prior investigations talked about in related work. the bogus positive rate is likewise low as displayed by utilizing both proposed approaches. We utilized precision and the blunder rate as

execution measures in our multiclass classifier review. Precision is the high of accurately grouped examples, and the misclassified is the small portion of misclassified occasions in a dataset. These two measures really sum up the general execution by considering of the classes and summing up the classifier execution as far as the union ways of behaving, accuracy, review, and F-measure displayed in Table 2. We tried the executed model in one of our assets on haze of Amazon Web Services (affirmation) gives us great execution in speed and time. We tested the model on 5 homogenous nodes deployed using the AWS and loaded the system with a random task creation and dynamic job ordering method, the workload was created dynamically for each data node, logs are collected and process on a separate node. This produced the quick result for finding the faulty nodes as the AWS node were highly configured. Figure 7 shows line chart for accuracy, pression, recall and F-measure for comparing result of different algorithm.

Table 1. Comparative analysis for identification of faulty nodes using ML

Approach	Classifier	Accuracy	False positive
Proposed	Hybrid method using ML	99.5%	2%
Method discussed in [12]	Random forest	96%	1.5%
Method discussed in [14]	K-NNSVM	92% 97%	8% 10%

Table 2. Faulty nodes detection on VMM (Virtual Machine Monitor)

Training data	Test data	Classifier	Accuracy	Precision	Recall	F-Measure
10%	10%	Hybrid method using ML	99.6456%	0.997	0.997	0.997
10%	10%	Random Forest	99.1095%	0.990	0.991	0.990

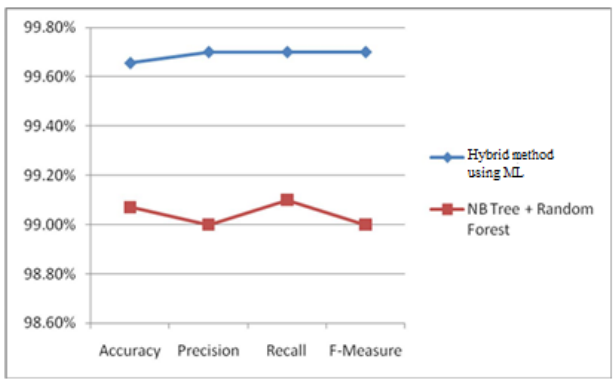

Figure 7. Resultant graph

In highlight development we use discretization cycle to change over the consistent space of an element into an ostensible area with a limited number of values. Front-end discretization is vital for certain classifiers on the off chance that their calculations can't deal with nonstop highlights by plan. While in back end for internet learning highlight decrease part of VMM diminishes the list of capabilities for production of ordinary model which makes the edge on moving normal base, so the new qualities might be much effective for direction. The oddity locator part decides the irregular and typical occasions by coordinating the gathered elements with the classes, in light of the recognized class the classifier chooses as typical or peculiarity occasion, at long last, the abnormality alert framework part produces a caution assuming oddity occasions are identified.

6. CONCLUSION

We described a system VM choosing and distribution problem using problem concepts and computational techniques. Given the massive number, we could not characterize all the features on this paper. Still, we did our best to give a representative sample of the most significant features. As we have seen, most features deal with both the single issue, but there are considerable variances in the problem formulations employed in each study, even within those two large clusters. At the moment, the research on these two fragments is generally fragmented, with just a few studies covering both. However, we proposed that, in the future, a confluence of these two domains would be required to represent hybrid cloud situations. The system evaluates virtual machines by using log data identification of heavy nodes. During the execution of the proposed approach, we evaluate the entire data using hybrid machine learning algorithm to get accurate predictions. In a distributed environment, this system is beneficial to detecting heated notes. Using this approach, we can quickly identify nodes that are already heated and select nodes that are in an ideal position. In our experiment, we demonstrate that the proposed system provides almost zero data leakage due to systematic management. To implement the proposed approach with various collaborative deep learning algorithms with additional local features will be the future work of the system.

REFERENCES

[1] Sakir, R.K.A., Bhardwaj, S., Kim, D.S. (2021). Enhanced faulty node detection with interval weighting factor for distributed systems. Journal of Communications and Networks, 23(1): 34-42. https://doi.org/10.23919/JCN.2021.000002

[2] Shial, R.K., Gouda, B.S., Pattanaik, S.R., Sethi, N. (2020). A centralized faulty node detection algorithm based on statistical analysis in WSN. In 2020 International Conference on Computer Science, Engineering and Applications (ICCSEA), pp. 1-6. https://doi.org/10.1109/ICCSEA49143.2020.9132847

[3] Sajan, S., Chacko, S.J., Pai, V., Pai, B.K. (2020). Performance evaluation of various algorithms that affect fault detection in wireless sensor network. In 2020 Fourth International Conference on Inventive Systems and Control (ICISC), pp. 540-545. https://doi.org/10.1109/ICISC47916.2020.9171070

[4] Niu, Y., Sheng, L., Gao, M., Zhou, D. (2021). Distributed intermittent fault detection for linear stochastic systems over sensor network. IEEE Transactions on Cybernetics, 52(9): 9208-9218. https://doi.org/10.1109/TCYB.2021.3054123

[5] Jia, W., Wang, J. (2020). Partial-nodes-based distributed

fault detection and isolation for second-order multiagent systems with exogenous disturbances. IEEE Transactions on Cybernetics, 52(4): 2518-2530. https://doi.org/10.1109/TCYB.2020.3007655

[6] Das, A., Mueller, F., Rountree, B. (2020). Aarohi: Making real-time node failure prediction feasible. In 2020 IEEE International Parallel and Distributed Processing Symposium (IPDPS), pp. 1092-1101. https://doi.org/10.1109/IPDPS47924.2020.00115

[7] Mizutani, K. (2021). Stateless node failure information propagation scheme for stable overlay networks. IEEE Access, 9, 88737-88745. https://doi.org/10.1109/ACCESS.2021.3090028

[8] Gaykar, R.S., Nalini, C., Joshi, S.D. (2021). Identification of straggler node in distributed environment using soft computing algorithms. In 2021 International Conference on Emerging Smart Computing and Informatics (ESCI), pp. 1-5. https://doi.org/10.1109/ESCI50559.2021.9396825

[9] Gaykar, R.S., Khanaa, V., Joshi, S.D. (2021). Detection of faulty nodes in distributed environment using machine learning. In 2021 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), pp. 228-232. https://doi.org/10.1109/ICAC3N53548.2021.9725478

[10] Numata, N., Ishigai, M., Tarutani, Y., Fukushima, Y., Yokohira, T. (2020). An IP fast reroute method against multiple node failures. In 2020 International Conference on Information and Communication Technology Convergence (ICTC), pp. 714-719. https://doi.org/10.1109/ICTC49870.2020.9289623

[11] Dhingra, M., Gupta, N. (2020). Failure node reduction algorithm to enhance fault tolerance capability of cloud nodes. In 2020 12th International Conference on Computational Intelligence and Communication Networks (CICN), pp. 165-168. https://doi.org/10.1109/CICN49253.2020.9242615

[12] Lu, J., Chen, L., Li, L., Feng, X. (2019). Understanding node change bugs for distributed systems. In 2019 IEEE 26th International Conference on Software Analysis, Evolution and Reengineering (SANER), pp. 399-410. https://doi.org/10.1109/SANER.2019.8668027

[13] Yarinezhad, R., Hashemi, S.N. (2019). Distributed faulty node detection and recovery scheme for wireless sensor networks using cellular learning automata. Wireless Networks, 25(5): 2901-2917. https://doi.org/10.1007/s11276-019-02005-7

[14] Bhaskar, S., Vijaya, C. (2019). Efficient protocol for node failure detection in wireless sensor network (WSN) to improve QoS. International Journal of Advanced Studies of Scientific Research, Forthcoming.

[15] Nikhil, C.S., Bollarapu, M.J., Neeraj, K.S.S., Kumar, A.P., Raghuveer, M.J.S.S. (2021). Efficient identification of node failure and recovery through end to end Probing techniques. In Journal of Physics: Conference Series, 2040(1): 012006. https://doi.org/10.1088/1742-6596/2040/1/012006

[16] Bangare, S.L. (2022). Classification of optimal brain tissue using dynamic region growing and fuzzy min-max neural network in brain magnetic resonance images. Neuroscience Informatics, 2(3): 100019. https://doi.org/10.1016/j.neuri.2021.100019

[17] Bangare, S.L., Pradeepini, G., Patil, S.T. (2006). Implementation for brain tumor detection and three dimensional visualization model development for reconstruction. ARPN Journal of Engineering and Applied Sciences (ARPN JEAS), 13(2): 467-473.

[18] Bangare, S.L., Dubal, A., Bangare, P.S., Patil, S.T. (2015). Reviewing Otsu's method for image thresholding. International Journal of Applied Engineering Research, 10(9): 21777-21783.

[19] Bangare, S.L., Pradeepini, G., Patil, S.T. (2018). Regenerative pixel mode and tumour locus algorithm development for brain tumour analysis: A new computational technique for precise medical imaging. International Journal of Biomedical Engineering and Technology, 27(1-2): 76-85. https://doi.org/10.1504/IJBET.2018.093087

[20] Bangare, S.L., Pradeepini, G., Patil, S.T. (2017). Neuroendoscopy adapter module development for better brain tumor image visualization. International Journal of Electrical & Computer Engineering (2088-8708), 7(6). https://doi.org/10.11591/ijece.v7i6.pp3643-3654

[21] Shelke, N., Chaudhury, S., Chakrabarti, S., Bangare, S.L., Yogapriya, G., Pandey, P. (2022). An efficient way of text-based emotion analysis from social media using LRA-DNN. Neuroscience Informatics, 100048. https://doi.org/10.1016/j.neuri.2022.100048

[22] Gupta, S., Kumar, S., Bangare, S.L., Nuhmani, S., Alguno, A.C., Samori, I.A. (2022). Homogeneous decision community extraction based on end-user mental behavior on social media. Computational Intelligence and Neuroscience, 2022: 1-9. https://doi.org/10.1155/2022/3490860

[23] Duy, T.V.T., Sato, Y., Inoguchi, Y. (2010). Performance evaluation of a green scheduling algorithm for energy savings in cloud computing. In 2010 IEEE international symposium on parallel & distributed processing, workshops and Phd forum (IPDPSW), pp. 1-8. https://doi.org/10.1109/IPDPSW.2010.5470908

[24] Kousiouris, G., Cucinotta, T., Varvarigou, T. (2011). The effects of scheduling, workload type and consolidation scenarios on virtual machine performance and their prediction through optimized artificial neural networks. Journal of Systems and Software, 84(8): 1270-1291. https://doi.org/10.1016/j.jss.2011.04.013

[25] Niehorster, O., Krieger, A., Simon, J., Brinkmann, A. (2011). Autonomic resource management with support vector machines. In 2011 IEEE/ACM 12th International Conference on Grid Computing, pp. 157-164. https://doi.org/10.1109/Grid.2011.28

[26] Xu, C.Z., Rao, J., Bu, X. (2012). URL: A unified reinforcement learning approach for autonomic cloud management. Journal of Parallel and Distributed Computing, 72(2): 95-105. https://doi.org/10.1016/j.jpdc.2011.10.003

[27] Beloglazov, A., Buyya, R. (2012). Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers. Concurrency and Computation: Practice and Experience, 24(13): 1397-1420. https://doi.org/10.1002/cpe.1867

[28] Pande, S.D., Chetty, M.S.R. (2018). Analysis of capsule network (Capsnet) architectures and applications. J Adv Res Dynam Control Syst, 10(10): 2765-2771.

[29] Pande, S.D., Chetty, M.S.R. (2019). Position invariant spline curve based image retrieval using control points. Int J Intell Eng Syst, 12(4): 177-191.

https://doi.org/10.22266/ijies2019.0831.17
[30] Pande, S.D., Patil, U.A., Chinchore, R., Chetty, M.S.R. (2019). Precise approach for modified 2 stage algorithm to find control points of cubic bezier curve. In 2019 5th International Conference on Computing, Communication, Control and Automation (ICCUBEA), pp. 1-8. https://doi.org/10.1109/ICCUBEA47591.2019.9128550
[31] Gaykar, R.S., Khanaa, V., Joshi, S.D. (2022). Faulty node detection in HDFS using machine learning techniques. Revue d'Intelligence Artificielle, 36(4): 553-560. https://doi.org/10.18280/ria.360406
[32] Gaykar, R.S., Khanaa, V., Joshi, S.D. (2022). A hybrid supervised learning approach for detection and mitigation of job failure with virtual machines in distributed environments. Ingénierie des Systèmes d'Information, 27(4): 621-627. https://doi.org/10.18280/isi.270412
[33] Bharathi, L., Chandrabose, S. (2022). Machine learning-based malware software detection based on adaptive gradient support vector regression. International Journal of Safety and Security Engineering, 12(1): 39-45. https://doi.org/10.18280/ijsse.120105
[34] Kokate, S., Chetty, M.S.R. (2021). Credit risk assessment of loan defaulters in commercial banks using voting classifier ensemble learner machine learning model. International Journal of Safety and Security Engineering, 11(5): 565-572. https://doi.org/10.18280/ijsse.110508