IJRAR.ORG E-ISSN: 2348-1269, P-ISSN: 2349-5138

INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS (IJRAR) | IJRAR.ORG

An International Open Access, Peer-reviewed, Refereed Journal

Optimizing Service Performance Through Hybrid Fog-Cloud Offloading

[1]Kiruthika D, [2]Raghul B, [3]Balamurugan S, [4]Manikandan S

[1]Assistant Professor, [2]UG Scholar, [3] UG Scholar, [4] UG Scholar

Information Technology,

Nandha Engineering College, Erode, India

Abstract : In this study, a Cloud system, Virtual Machines (VMs) are scheduled to hosts according to their instant resource usage (e.g., to hosts with most available RAM) without considering their overall and long-term utilization. Also, in many cases, the scheduling and placement processes are computational expensive and affect performance of deployed VMs. In this work, a Cloud VM scheduling algorithm that considers already running VM resource usage over time by analyzing past VM utilization levels to schedule VMs by optimizing performance by using KNN and Naive Bayes classification technique. The Euclidean distance of KNN is measured and then virtual machine is scheduled on the physical machine. The Cloud management processes, like VM placement, affect already deployed systems so the aim is to minimize such performance degradation. Moreover, overloaded VMs tend to steal resources from neighboring VMs, so the work maximizes VMs real CPU utilization. The results show that our solution refines traditional Instant-based physical machine selection as it learns the system behavior as well as it adapts over time. The concept of VM scheduling according to resource monitoring data extracted from past resource utilizations (including PMs and VMs). The count of the physical machine gets reduced by four using K-NN & NB classifier.

***Keywords*—Cloud Data Center, Virtual Machine, Energy Consumption, Resource Management, Task Scheduling.**

I. INTRODUCTION

The rapid proliferation of new digital devices, leveraging emerging networking technologies, has led to the generation of increasingly sophisticated tasks. These tasks, often manifested as online applications, place significant demands on processing power, data rates, and various resources [1]–[3]. However, despite the advancements in device design, contemporary devices often struggle to run resource-intensive applications like virtual reality, smart healthcare, and select Internet of Things (IoT) applications. In contrast to traditional computing platforms, IoT nodes exhibit limited capabilities due to their inherent functionalities in specialized environments such as factories and hospitals. To address these limitations, researchers have proposed the concept of task offloading, allowing resource-intensive applications to be executed remotely on high-performance entities. This approach is particularly relevant for scenarios where IoT nodes lack sufficient processing capabilities. Task offloading involves the execution of applications on fog nodes or cloud servers, thereby alleviating the computational burden on end-nodes. Fog nodes or cloud servers execute these offloaded applications on behalf of end-nodes, enabling the deployment of resource-intensive applications in resource-constrained environments. In cloud computing has emerged as a prominent solution for executing high-complexity tasks across diverse domains. Cloud services offer extensive capabilities such as large storage, high-speed communication, and powerful processing units. Cloud computing services are accessible online from anywhere, attracting a significant user base. However, the geographical distance between edge-node devices or IoT devices and the cloud can introduce significant latency, particularly for real-time applications sensitive to delay. To mitigate this challenge, fog computing has emerged as an intermediary layer between cloud servers and edge-node devices. Fog nodes, positioned closer to end-nodes, facilitate quicker access to resources and reduce latency, thereby enhancing the performance of IoT applications.

In our project, we are focused on Virtual Machine (VM) scheduling, which is crucial for making the best use of resources and meeting Quality of Service (QoS) requirements. VM scheduling involves assigning VM instances to physical resources like servers or fog nodes, considering factors such as resource availability, workload, and performance goals. To do this efficiently, we are proposing to use machine learning algorithms, specifically K-nearest neighbors (KNN) and Naive Bayes (NB). KNN is a simple yet effective classification algorithm.

It works by finding the K-nearest data points to a given query point and assigning a label based on the majority class among them. In VM scheduling, KNN helps predict the best placement for VMs by analyzing historical patterns. On the other hand, NB is a probabilistic classifier that relies on Bayes' theorem with a "naive" assumption of feature independence

Our project aims to develop a VM scheduling framework that leverages the predictive capabilities of KNN and NB algorithms to optimize resource allocation and enhance QoS in fog computing environments. By integrating machine learning techniques into VM scheduling, we seek to improve the efficiency and effectiveness of resource management, ultimately enhancing the performance of applications.

1.1 CLOUD DATA CENTERS

Cloud data centers have become the backbone of contemporary computer infrastructure in the age of rapid digital transformation, completely changing how consumers and organizations access and manage data. These centers mark a significant transition from conventional on-premises data processing and storage to remotely hosted, scalable, and adaptable computing environments. Cloud data centers offer a wide range of services, from networking and analytics to storage and processing, allowing businesses to dynamically scale their resources in response to demand. The built-in benefits of cost-effectiveness, scalability, and accessibility have rendered cloud data centers essential in today's globalized society.

1.2 VIRTUAL MACHINE

Several operating systems (OS) can run on a single physical machine thanks to the virtual machine (VM), a software-based simulation of a real computer. With the use of this technology, separated environments that function independently of the underlying hardware also referred to as virtualized instances, or VMs can be created for operating systems and applications. A hypervisor, also known as a Virtual Machine Monitor (VMM), is a crucial part of virtual machines since it controls and distributes the host machine's physical resources to the virtual machines. Two varieties of hypervisors exist: Type 2 (hosted) hypervisors operate on top of an already-installed operating system, whereas Type 1 (bare-metal) hypervisors operate directly on the hardware. Virtual machines (VMs) find extensive application in several computer contexts, including cloud computing, testing and development environments, and server consolidation.

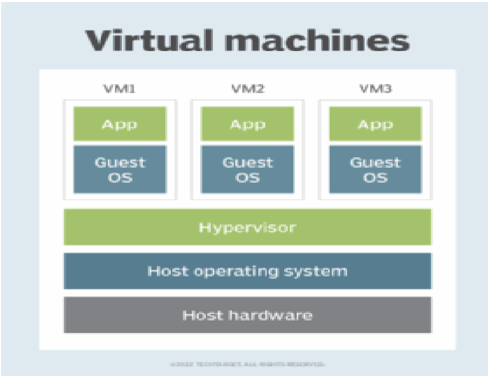

Fig.1 Virtual Machines

1.3 ENERGY CONSUMPTION

Energy consumption is one of the most important worldwide factors in the modern technological and industrial landscape. The need for energy is growing as societies depend more and more on cutting-edge technologies to run their daily lives. The difficulty is not only supplying the energy required to power homes, businesses, and the extensive network of data centers that supports our digital infrastructure, but also doing so in a sustainable manner. Energy use's effects on the environment have come to be of great concern, especially when it comes to computing and data processing. The energy consequences of these breakthroughs must be understood and addressed as the world moves toward increasingly digital and networked systems. This introduction lays the groundwork for a discussion of the intricacies surrounding energy use, highlighting the urgent need for creative frameworks and solutions that support sustainability and efficiency across a range of industries.

1.4 RESOURCE MANAGEMENT

The Effective resource management is indispensable for the smooth functioning of various sectors, including computing systems and corporate enterprises. It involves strategically allocating, utilizing, and enhancing resources such as financial assets, technological infrastructure, and human capital. Particularly in the realm of technology and computing, efficient resource management is paramount. As organizations navigate the complexities of the digital age, the ability to judiciously distribute computing resources becomes crucial for maintaining optimal performance and adaptability to changing demands. With the rise of cloud computing, resource management gains even greater significance, necessitating innovative approaches for handling both virtual and physical assets. This includes intelligently orchestrating resources, dynamically provisioning computing instances, and optimizing workloads to maximize efficiency while minimizing costs. In response to dynamic workloads, fluctuating demands, and evolving technologies, resource management extends beyond mere allocation. It encompasses proactive capacity planning, predictive analytics, and continuous optimization to ensure resources are utilized effectively in alignment with organizational objectives and performance targets. By embracing a holistic approach to resource management, organizations can fully leverage their technological ecosystem, driving innovation, agility, and competitiveness in an increasingly digital landscape.

1.5 TASK SCHEDULING

A key component of computer systems is task scheduling, which includes the planned arrangement and carrying out of different operations to maximize output and efficient use of resources. Effective task scheduling guarantees that computational workloads are assigned to available resources in a way that maximizes throughput, reduces delay, and improves overall system efficiency, whether in conventional computing environments or contemporary distributed systems. Decisions on when and where to complete activities are made during this process, keeping in mind resource constraints, dependencies, and priority. Effective task scheduling is crucial for achieving scalability, responsiveness, and cost-effectiveness in computing systems, from operating systems managing local tasks on a single machine to complex cloud computing scenarios where tasks may be distributed across a network of interconnected nodes.

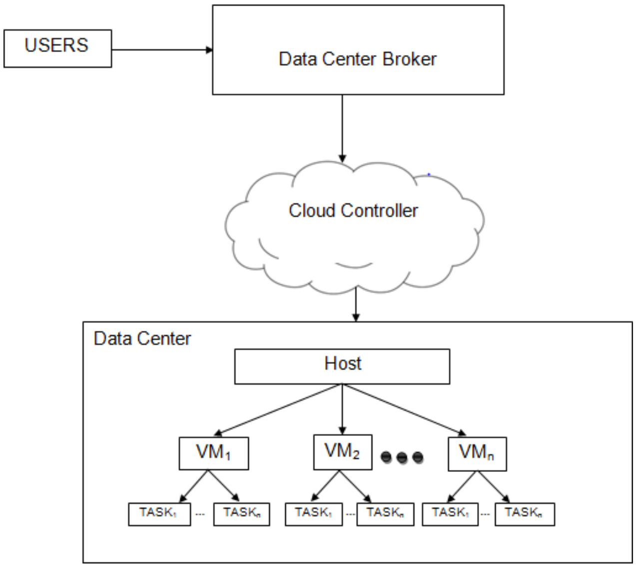

Fig.2 Task Scheduling

II. RELATED WORKS

A novel approach to remote data integrity checking (RDIC) is proposed in this paper by Yong Yu [1] et.al. The proposed identity-based (ID-based) RDIC protocol utilizes key-homomorphic cryptographic primitive to reduce the complexity of the system and the cost of establishing and managing the public key authentication framework in PKI-based RDIC schemes. The security model of the proposed ID-based RDIC protocol is formalized, including security against a malicious cloud server and zero knowledge privacy against a third-party verifier. The protocol is proven to be secure against the malicious server in the generic group model and achieves zero knowledge privacy against a verifier. Extensive security analysis and implementation results demonstrate that the proposed protocol is both provably secure and practical for real-world applications**.**

This paper explores the challenges and models proposed for SLA [2] in cloud computing. Cloud computing offers distributed resources and on-demand services to organizations globally, but there are various challenges that exist in cloud services. To overcome these challenges, different techniques have been proposed, including models for SLA in cloud computing. We review the different models proposed for SLA in different cloud service models like SaaS, PaaS, and IaaS, and discuss their advantages and limitations. Additionally, we examine the role of the cloud service provider in establishing SLA and the parameters that consumers must consider before signing SLA in the cloud platform. Overall, this survey provides insights into the challenges and solutions for SLA in cloud computing**.**

PritiNarwal [3] et.al has proposed a paper on the topic of Cloud Computing, which is a dynamic platform that utilizes virtualization technology. In a Cloud computing environment, virtualization abstracts the hardware system resources in software, enabling each application to run in an isolated environment known as the virtual machine. The allocation of virtual machines to different users on the same server is done by the hypervisor. While Cloud computing offers numerous benefits such as resource-sharing, cost-efficiency, high-performance computability, and reduced hardware costs, it also poses several security threats. These threats can directly impact Virtual Machines (VMs) or indirectly affect the hypervisor through the virtual machines hosted on it. This paper provides a comprehensive review of all potential security threats and proposes countermeasures using Game Theoretic approaches. Game Theory is employed as a defensive measure due to the independent and strategic decision-making nature of cloud users, where each player competes for the best possible secure solution. In addition to security and privacy concerns, it is crucial to address other issues like efficiency and optimization, considering that different users have varying resource requirements in a cloud environment.

Nitin Kumar Sharma [4] et.al has proposed a paper that introduces Attribute Based Access Control (ABAC) models as a solution to the limitations of classical access control models (DAC, MAC, and RBAC) while incorporating their advantages. ABAC provides access control based on generic attributes of entities, aligning with many organizational security policies that rely on attributes for access decisions. The paper utilizes the Web Ontology Language (OWL) to formally define and process security policies, enabling the use of a reasoned to determine access permissions. The ABACα model, represented in OWL, is presented to enforce policies using the EYE reasoned, which infers logical relationships and grants access for requested actions. This paper serves as an initial step towards specifying and enforcing machine understandable policies within the ABAC model, which is recognized as one of the most comprehensive access control models available today. Additionally, the paper acknowledges the need for further analysis on the performance of the reasoning process and highlights the limitations of the basic ABACα model, such as the absence of static/dynamic separation of duties and the lack of additional attributes for contextual information.

In this study, Ziad Ismail [5] et.al proposed a paper that addresses the security challenges introduced by new developments in cloud computing. The focus is on ensuring the confidentiality, integrity, and availability of outsourced data. To achieve this, a Service Level Agreement (SLA) is typically signed between the cloud provider and the customer. One important aspect of the SLA is verifying the cloud provider's compliance with data backup requirements for redundancy purposes. There are various security mechanisms available to check the integrity and availability of outsourced data. This task can be performed by the customer or delegated to an independent entity referred to as the verifier. However, frequent data verification can lead to additional costs, which may discourage customers from performing it regularly. To address this, we propose using game theory to capture the interaction between the verifier and the cloud provider.

III. EXISTING SYSTEM

Without a cord Through the internet, cloud computing provides data and computer resources on a pay-per-use basis. This allows us to update our software automatically. Our carbon impact is decreased because we can just use the space needed for the server. The primary issue in cloud computing that lowers system performance is task scheduling. An effective task-scheduling method is required to increase system performance. Current taskscheduling algorithms prioritize considerations such as CPU memory, task resource requirements, execution time, and execution cost. They do not, however, take network bandwidth into account. In this paper, we provide an effective task scheduling technique that takes network capacity into account to provide divisible task scheduling. This allows us to distribute the workflow according to the amount of network bandwidth that is available. Our suggested task-scheduling approach allocates the appropriate number of tasks to each virtual machine using a nonlinear programming model for divisible task scheduling. We create a divisible load scheduling technique based on the allocation, taking network bandwidth into account.

IV. PROPOSED SYSTEM

The suggested system solves the drawbacks of the existing instant-based resource allocation and presents a novel method for scheduling Cloud Virtual Machines. Utilizing past information on virtual machine resource usage, the system uses a scheduling method driven by KNN with NB classifier. Based on past performance data, KNN is a well-liked machine learning method for regression and classification tasks. It is also used to forecast the processing capacity of the server. Conversely, NB is a straightforward and effective text classification technique that assumes that each feature is independent. Through time, the system may learn from and adjust to the dynamic behavior of the cloud environment thanks to this methodology. The suggested methodology, in contrast to conventional methods, places an emphasis on long-term and total resource consumption with the goal of reducing the influence of Cloud management procedures on deployed virtual machines. The system provides improved efficiency and increases real CPU utilization by optimizing performance and minimizing the number of physical machines through the KNN with NB classifier. This helps to improve the traditional VM placement tactics in Cloud systems.

1.1 VM SCHEDULING

The main element in charge of coordinating the distribution of virtual machines in the cloud environment is the VM Scheduling module. To decide where to deploy virtual machines (VMs), it interfaces with system parameters, user demands, and historical data. This module employs the KNN with NB classifier to forecast server processing capability and improve virtual machine placement, drawing on insights from the Data Analysis module. It seeks to increase the effectiveness of Cloud system administration, lessen the requirement for instantaneous resource allocation, and improve overall resource consumption.

1.2 DATA ANALYSIS

The system is built upon the Data Analysis module. It gathers, organizes, and examines past information about virtual machine resource usage. The system can recognize patterns and trends in resource demands over time because to this data-driven methodology. The module provides input to the VM Scheduling module by using statistical and machine learning techniques to derive relevant insights. The system becomes adaptive to the dynamic nature of the Cloud environment by continuously learning from prior performance, which helps with more precise decision-making.

1.3 KNN WITH NB CLASSIFIER

K-Nearest Neighbors (KNN) and Naive Bayes (NB), two potent machine learning techniques, are combined in the KNN with NB Classifier module. Based on past performance data, KNN is used to forecast server processing capacity. In order to forecast future events, it finds commonalities between historical and present patterns of resource consumption. However, NB, a text classification algorithm, assumes that each feature is independent and adds to the overall classification scheme. The system's capacity to allocate resources and locate virtual machines (VMs) more intelligently is improved by the combination of these techniques.

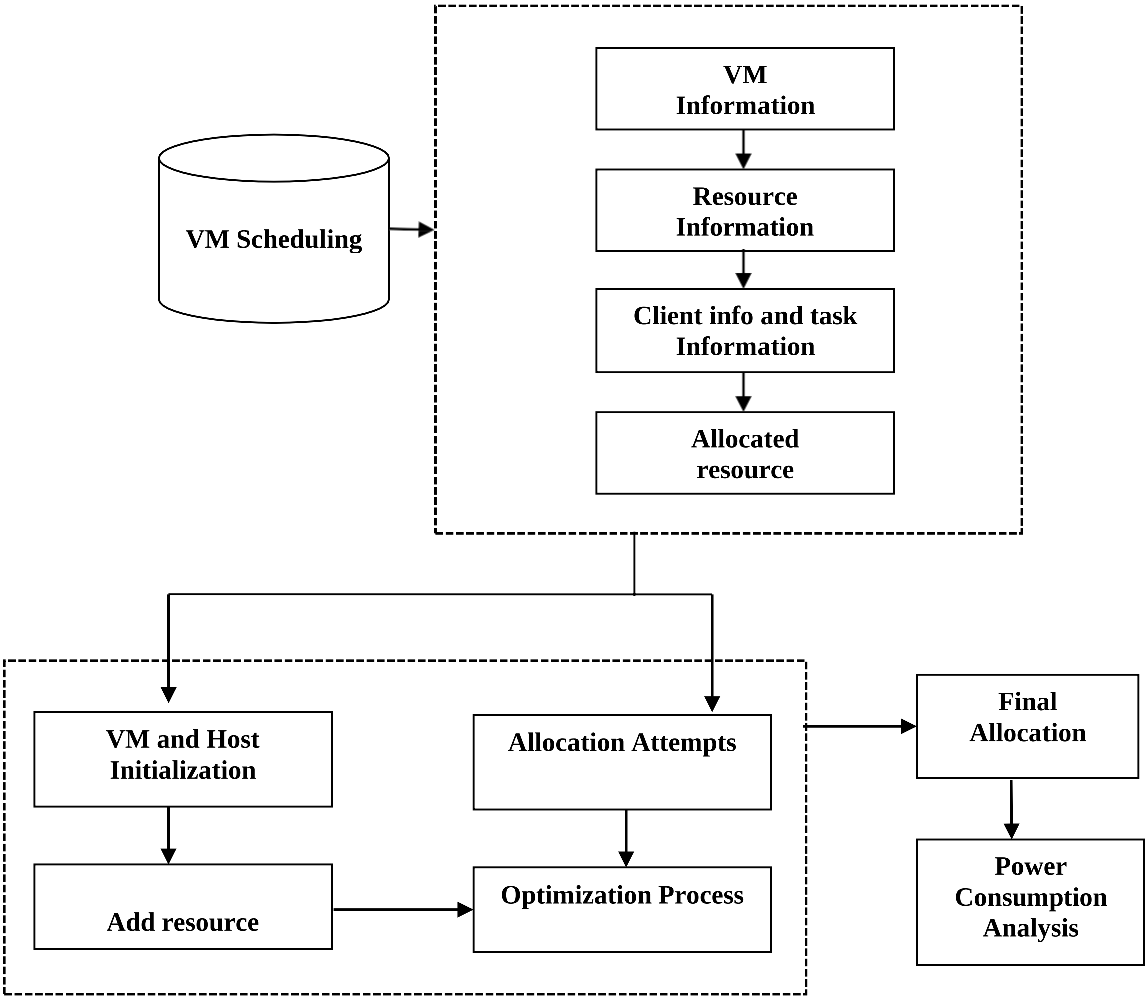

Fig.3 Workflow Diagram

1.4 OPTIMIZATION SCHEME

The Optimization Scheme module works to improve the traditional methods for placing virtual machines (VMs) in cloud systems. Through the utilization of KNN with NB Classifier module insights, this component seeks to minimize the number of physical computers while optimizing performance. The goal of the optimization method is to reduce the effect of cloud management procedures on deployed virtual machines (VMs) while optimizing real CPU consumption. By applying smart and data-driven decision-making techniques, this module is essential to achieving increased efficiency in Cloud systems.

V. ALGORITHM DETAILS

5.1 KNN CLASSIFIER

In pattern recognition, the K-Nearest Neighbors (KNN) algorithm is a non-parametric method utilized for classification and regression tasks [2], [6]. KNN operates on the principle of proximity in feature space, where objects are classified based on their similarity to other data points. This proximity is typically measured using a distance metric, such as the Euclidean distance. For classification tasks, the KNN algorithm assigns a class label to a query point based on the class labels of its (K) nearest neighbors in the training dataset. Mathematically, let (xq) be the query point, (xi) be the ith point in the training dataset, (yi) be the corresponding class label, and d (xq, xi) be the distance between (xq) and (xi). The class label of the (xq) is determined by a majority vote among its (K) nearest neighbors:

$$\hat{y}_q = \arg max_y \sum_{i=1}^{K} I(y_i = y)$$

where ($\hat{y}q$) is the predicted the class label for an (xq), I(.)is the indicator function, and (y) ranges over all possible class labels. Similarly, for regression tasks, the KNN algorithm predicts the value of a query point based on the average value of its (K) nearest neighbors' target values. Mathematically, let (xq) be the query point, (xi) be will the ith point in the training a dataset, (yi) be the corresponding target value, and d (xq, xi) be the distance between (xq) and (xi). The predicted value of (xq) is given by:

$$\hat{y}_q = \frac{1}{k}\sum_{i=1}^{k} y_i$$

where ($\hat{yq}$) is the predicted value for (xq). For example, consider a dataset containing information about various fruits, with features such as weight and texture, and corresponding labels indicating the type of fruit (e.g., apple, banana, orange). By applying the KNN algorithm to this dataset, one can classify new fruits based on their similarity to existing fruits in the dataset, thereby enabling accurate identification of unknown fruits. Similarly, in a regression scenario, the KNN algorithm can predict the price of a house based on the prices of similar houses in the vicinity.

The K-nearest neighbors (KNN) algorithm calculates distances between data points using various distance metrics, such as Euclidean distance, Manhattan distance, or cosine similarity. For Euclidean distance, the formula to calculate the distance (d) between two points (x_1, y_1) and (x_2, y_2)in a two-dimensional space is:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

In a higher-dimensional space, the formula extends accordingly. This formula calculates the straight-line distance between two points in the feature space. In the context of KNN, this distance is computed between the query point and each of the training data points to find the K-nearest neighbors.

5.2 NAVIE BAYES ALGORITHM

The Naïve Bayes algorithm is a straightforward probabilistic classification technique that derives its probability values by computing frequency and value combinations from the associated dataset. It operates under the assumption that all attributes are independent of each other. The classification process of Naïve Bayes involves analyzing various clues or directions to determine the class of the data being evaluated. Input: Training dataset (T). Predictor variables in the testing dataset F = ($f_{1,} f_{2,} f_{3,} \ldots, f_n$). Output: Class of the testing dataset.
Algorithm Steps:

- Read the training dataset (T).
- Calculate the mean (μ) and standard deviation (σ) of the predictor variables in each class.
- Repeat the following steps:
- Calculate the probability of f_i using the Gaussian density equation in each class until the probability of all predictor variables ($f_{1,} f_{2,} f_{3,} \ldots, f_n$) has been calculated.
- Calculate the likelihood for each class.
- Determine the class with the highest likelihood.

The Gaussian density equation, also known as the probability density function (PDF) of the normal distribution, is used to calculate the probability of a given value x for a predictor variable f_i in a particular class. Mathematically, it is represented as:

$$P(x|\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e - \frac{(x-\mu)^2}{2\sigma^2}$$

Where:

- (μ) is the mean of the predictor variable of f_i in the class.
- (σ) is the standard deviation of the predictor variable f_i in the class.
- (e) is the base of the natural logarithm (approximately 2.71828).

This equation computes the probability of observing the value (x) given the mean μ and standard deviation (σ) of the predictor variable f_i in a particular class, assuming a normal (Gaussian) distribution. The Naïve Bayes algorithm applies this probability calculation for each predictor variable f_i in each class to determine the likelihood of each class for a given instance. The class with the highest likelihood is then assigned to the instance being classified.

VI. RESULT ANALYSIS

The accuracy comparison between the Non-Linear Programming Model and the KNN with NB algorithm is summarized in the table below:

Algorithm	**Accuracy**
Non-Linear Programming Model	75
KNN with NB	81

Table.1 Comparison Table

These values represent the accuracy rates achieved by each algorithm in forecasting and optimizing virtual machine resource allocation. The KNN with NB algorithm outperforms the Non-Linear Programming Model, demonstrating a higher accuracy of 81% compared to 75%.

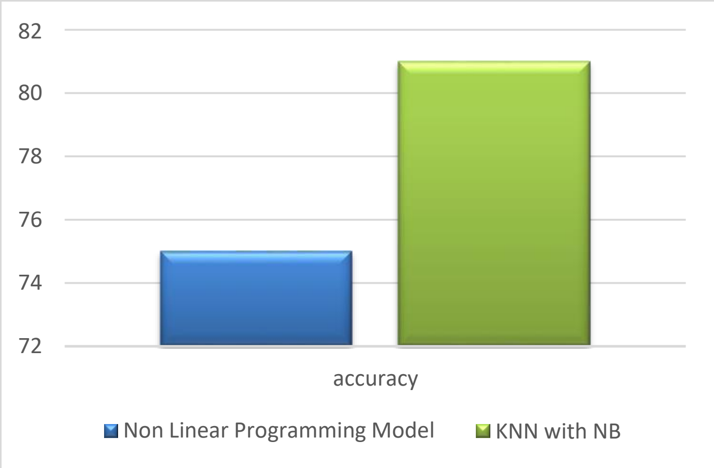

Fig.4 comparison Graph

In system evaluation, the Non-Linear Programming Model achieved a 75% accuracy rate in forecasting VM resource allocation, while the KNN with NB algorithm performed better at 81%. This indicates the superiority of the machine learning-driven approach, leveraging historical data and KNN with NB classifier, in enhancing VM scheduling accuracy. The KNN with NB system's increased precision suggests better forecasting of server processing capability from past data, leading to improved resource utilization and VM placement. These metrics underscore the potential for more efficient Cloud VM scheduling and better decision-making compared to conventional methods.

VII. CONCLUSION

Finally, by combining historical data analysis and machine learning algorithms specifically, KNN with NB Classifier the suggested VM scheduling system exemplifies a progressive approach to Cloud resource management. This optimizes VM placement and boosts overall efficiency. The approach differs from conventional instant-based allocation techniques in that it places a greater emphasis on long-term resource usage, adaptation to dynamic Cloud settings, and the reduction of management process consequences. The system attempts to offer users a dependable and intelligent way to make sense of virtual machine allocation through thorough testing and careful implementation.

Using past performance data and pattern recognition, the system can help increase CPU usage, decrease the number of physical machines in Cloud systems, and improve scalability. It is anticipated that the successful deployment and operation of this novel virtual machine scheduling system will greatly improve the efficiency and long-term viability of cloud resource allocation techniques.

VIII. FUTURE WORK

The suggested VM scheduling mechanism offers opportunities for future improvement and expansion. Exploring additional machine learning techniques or advanced predictive models can enhance the precision of resource usage forecasts. This includes delving into ensemble learning methods or deep learning architectures to extract nuanced patterns from historical data, refining forecasting accuracy. Additionally, investigating reinforcement learning techniques can optimize adaptability to workload changes, allowing real-time learning and adaptation. This interaction with the environment enhances responsiveness to sudden workload fluctuations, ensuring efficient resource allocation and performance optimization.

The dynamic nature of cloud technologies requires system expansion for multi-cloud support, enabling seamless resource distribution. Developing interoperability frameworks and orchestration mechanisms can enhance efficiency across heterogeneous cloud environments. Overall, these enhancements allow the VM scheduling mechanism to evolve and meet evolving cloud computing demands.

IX. REFERENCE

[1] Identity-based remote data integrity verification for cloud storage with perfect data privacy preservation, Y. Yong, M. H. Au, and G. Ateniese, IEEE Transactions on Information Forensics and Security, vol. 12, no. 4, pp. 767–778, 2017.

[2] Service level agreement in cloud computing: A survey, U. Wazir, F. G. Khan, and S. Shah, International Journal of Computer Science and Information Security, vol. 14, no. 6, p. 324, 2016.

[3] A study of game-theoretic methods for safe virtual machine resource allocation in the cloud by P. Narwal, D. Kumar, and M. Sharma, Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies, 2016.

[4] In the IEEE Tenth International Conference on Semantic Computing (ICSC),2016, pp. 333–336, N. K. Sharma and A. Joshi describe attribute-based access control mechanisms in Owl.

[5] Auditing a cloud provider's adherence to data backup requirements: A game theoretical approach, Z. Ismail, C. Kiennert, J. Leneutre, and L. Chen, IEEE Transactions on Information Forensics and Security, vol.11, no. 8, pp. 1685–1699, 2016.

[6] In Computer Standards & Interfaces, vol. 38, pp. 44–50, 2015, E. Furuncu and I. Sogukpinar published Scalable Risk Assessment Method for Cloud Computing Using Game Theory (CCRAM).

[7] Identity-based encryption with outsourced revocation in cloud computing, J. Li, J.W. Li, and X. F. Chen, IEEE Transactions on Computers, vol. 64, no. 2, pp. 425–437, 2015.

[8] Privacy-preserving association rule mining in cloud computing by X. Yi, F. Y. Rao, and E. Bertino, ACM Symposium on Information, Computer, and Communications Security 2015, pp. 439–450.

[9] In the Proceedings of IJCAI 2015, pp. 596–602, J. Lou and Y. Vorobeychik present equilibrium analysis of multi-defender security games.

[10] 2014 saw M. Nabeel and E. Bertino publish a paper in IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 9, pp. 2268–2280, on privacy delegated access control in public clouds.