USE OF CLUSTERING MACHINE LEARNING ALGORITHMS IN FOG COMPUTING FOR TASK SCHEDULING AND RESOURCE ALLOCATION

Suraj Rajaram Nalawade[1]*, Prof. Ashok Kumar Jetawat[2]

Abstract

In Fog computing environment the major issue is task scheduling and resource allocation. The study tries to address the above-mentioned issue using clustering machine learning algorithms like Canopy Clustering, Hierarchical Clustering and Make Density Based Clustering. The major objectives being associated with the research work were firstly to do comparative analysis of various clustering algorithms based on performance measures overall accuracy and to identify the most suitable clustering algorithm for resource allocation and task scheduling. The hypothesis testing results confirms that there is significant difference between various clustering algorithms based on performance measure overall accuracy at the overall accuracy values varies highly, the accuracy value of Canopy Clustering, Hierarchical Clustering and Make Density Based Clustering were found to be 74.5%, 59% and 48.5% respectively. Among all the clustering algorithms the most appropriate one for task scheduling and resource allocation is found to be Canopy Clustering.

[1]*Research Scholar, Pacific Academy of Higher Education and Research University, Udaipur, (Rajasthan)
[2]Professor, Pacific Academy of Higher Education and Research University, Udaipur (Rajasthan)

***Corresponding Author: -** Suraj Rajaram Nalawade
*Research Scholar, Pacific Academy of Higher Education and Research University, Udaipur, (Rajasthan)

DOI: 10.48047/ecb/2023.12.si5a.0399

1. Introduction

Large amounts of data are produced by fog computing systems, therefore more apps and services are being created. Robotics, neuromorphic computing, computer graphics, natural language processing (NLP), decision-making, and speech recognition have all seen substantial advancements thanks to machine learning (ML), a crucial field. Research into fog computing issues has been suggested using ML. Resource management, security, latency reduction, energy savings, and traffic modelling are just a few of the fog services that have benefited from the increased usage of machine learning (ML) in fog computing applications. To our knowledge, no research has looked at the function of ML in fog computing. As a result, our work shed light on ML functions for fog computing. Applications for ML fog computing offer deep insights and cleverer task responses. The paper looks at the most recent ML techniques for accuracy, security, and resource management in fog computing.

2. Background

In distributed systems, there are many different scheduling algorithms, and their major goals are to achieve high computing performance and improved system execution. To offer scheduling in a fog environment, conventional task scheduling techniques are incompatible. In order to optimize the execution of requests in fog networks, scheduling takes into account a number of factors, including task queue, request time, VM speed, and queue priority.

The bundled tasks are assigned to virtual computers using the Ant colony method. The technique is composed of three main steps: grouping activities according to cost and time; prioritizing tasks; and choosing the best virtual machine with the aid of the Ant colony algorithm. The simulation results are improved by this algorithm's low energy usage in iFogsim. [1]

According to Yin et al. (2018) proposes a method for optimizing process scheduling. This technique uses a tree-based primary structure to arrange procedures in a foggy environment. The resource discovery algorithm is used in the first phase to categories all available resources, after which each source transmits its information to the parent node that is directly connected to it, facilitating resource control by any parent node. In the second phase, resources are assigned to workflow tasks while taking the user-determined service quality into account. In general, this approach offers a way to schedule tasks while taking the client-specified QOS settings into account. [2]

Based on the least completion time (MCT), researchers have created the knapsack to schedule tasks for simultaneous video transfers in the cloud. The MCT algorithm was used for scheduling, after the max min technique was used to determine the most powerful computers and allocate them to a set of partitions. According to the results, the max min algorithm is superior than the MCT method in terms of both execution time and the number of segments. [3]

3. Methodology

We can now take advantage of the highly automated and performance-based system, on-demand services, and reduced operational costs in the IT market that have long been sought after thanks to the advent of cloud computing, to name just a few of the many benefits of this revolutionary technology. The system is user-driven and operates on a Service Oriented Architecture (SOA) to provide the services that customers need. Due to the service-oriented nature of cloud computing, individual must rely on a third party to store and show our data and information in servers and data centres located in secretive locations throughout the world. [4]

Cloud computing, the central server handles all of the processing of the data acquired by the node. This takes a lot of time since data must first be sent from the node to the central server before the server can process the data. Terabytes of data cannot be sent back and forth from the node to the cloud since it is not practicable. Fog computing, a cloud computing extension, is presented to address these drawbacks. In this, data processing is completed in full in the node if the data does not require more processing capacity and partially in the node if it does, with the data then being transmitted to the central server for further calculations. As a result, the procedure takes much less time and is more effective because the central server is not overworked. In geographically remote places with inconsistent connection, fog is quite helpful. Fog computing promises to meet the ideal use case's requirement for intelligence close to the edge, where ultra-low latency is crucial.

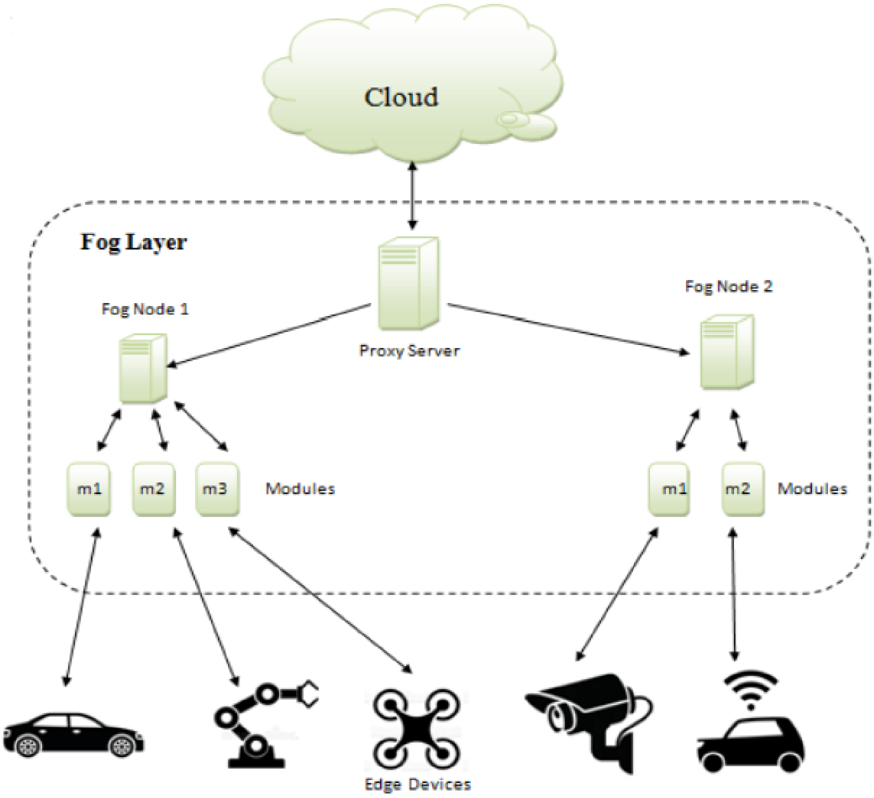

Figure 3.1: Fog-Cloud System [5]

Objectives:

1. Comparative analysis of clustering algorithms based on various performance measures overall accuracy.
2. To identify the most suitable clustering algorithm for resource allocation and task scheduling.

Hypotheses:

H_0 1: There is no significant difference between various clustering algorithms based on performance measure overall accuracy.

H_a 1: There is significant difference between various clustering algorithms based on performance measure overall accuracy.

iFogsim is a java-based simulator where we can create and test scenarios, topologies, and applications for fog computing. To test fog nods, sensors, and actuators in a fog environment, the create application function of the Fog Broker class first builds a fog environment. After that, it constructs a case study or any other application. The create Fog Device function generates a number of fog devices with various capacities and features. The suggested system is implemented with the aid of CloudSim 3.0 and IfogSim. To select a machine learning clustering algorithm WEKA tool is being used on a PC running Windows 7, the simulation is carried out.

Dataset Description:

The dataset consists of attribute values of 13 nodes among which 10 were the Fog nodes and remaining 3 were the cloud nodes. Two files based on number of tasks were being considered the first file consists of 40 tasks and the second file consists of 160 tasks. The various attributes being included were execution cost, delay, execution time, CPU usage, network usage etc.

4. Result and Discussion

Numerous advantages come with cloud computing, such as enormous processing power, plenty of storage, a vast network linking processing nodes and data sources, and a pay-per-use model. These paradigms are offered by cloud computing, a powerful technology that also offers many other advantages including flexibility, lower prices, scalability, and simplicity of installation. Nevertheless, despite these advantages, there are some drawbacks to cloud computing. The client and Cloud layer could be geographically apart, which could result in transmission delays; there might not be enough resources to complete tasks; many resources might be idle even if activities need to be completed immediately; and so on. To address these problems, virtualized fog computing technology is deployed. Between end users and cloud data centers lies a layer called fog.

Depending on where the data producer is located, fog computing may be effective for executing applications that need low latency and immediate replies. To handle incoming requests, this layer may include several virtual servers. The systematic process of assigning resources to the required Cloud customers over the Internet is known as resource allocation.
The study includes:

1. Analysis of Clustering based Scheduling in fog computing.
2. Implementation of proposed algorithm in iFogsim.
3. Reduction of Execution Cost.

Canopy Clustering:

Table 4.1: Performance Measure Canopy Clustering Based on Overall Accuracy

Measures	Values
Correctly Classified Instances	149 (74.5%)
Incorrectly Classified Instances	51 (25.5%)
Overall Accuracy	74.5%
Total Number of Instances	200
Time taken to build model	0.001 seconds

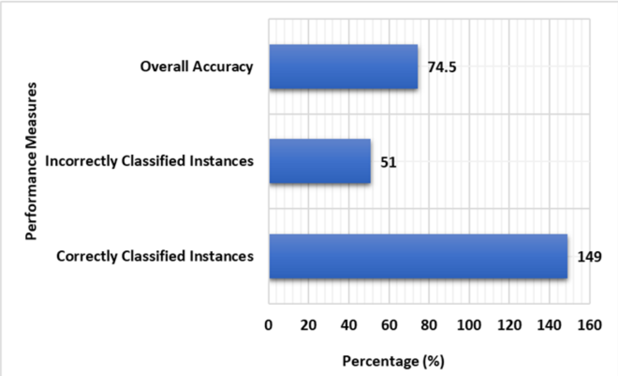

Figure 4.1: Overall Accuracy of Canopy Clustering

The results obtained by simulation confirms that correctly classified instances in case of Canopy Clustering were found to be 149 which clearly reflects that the overall accuracy shown by the Canopy Clustering is higher with 74.5% as compared to other clustering algorithms such as Hierarchical Clustering and Make Density Based Clustering. Similarly, the model building time of 0.001 seconds is quite less as compared with other algorithms also the recall, F-measure and precision values of 0.70,0 .70 and 0.75 respectively were higher as compared to Hierarchical Clustering and Make Density Based Clustering.

Table 4.2: Canopy Clustering Node or Class wise Results

S. No.	n (truth)	n (classified)	Accuracy	Precision	Recall	F1 Score	Class
1	86	80	0.77	0.75	0.70	0.72	Node1
2	30	40	0.76	0.28	0.37	0.31	Node2
3	44	40	0.96	0.95	0.86	0.90	Node3
4	40	40	1	1	1	1	Node4

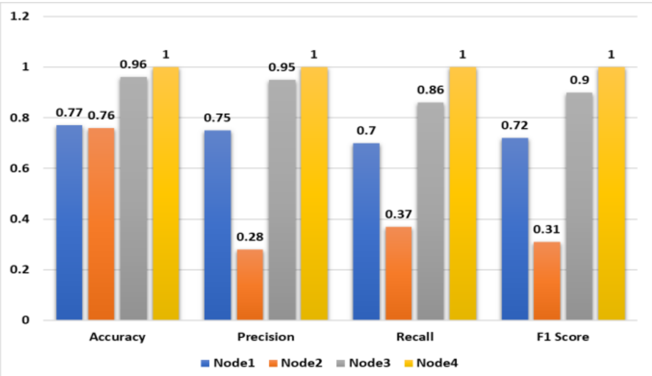

Figure 4.2: Canopy Clustering Node or Class wise Results

The figure above shows the node wise results. The node 4 shows highest accuracy, precision, recall and F-measure score.

Hierarchical Clustering:

The Hierarchical Clustering algorithm was being analysed for task scheduling and resource allocation in Fog environment. Accordingly, the results are shown below in the table and figure.

Table 4.3: Performance Measure Hierarchical Clustering Based on Overall Accuracy

Measures	Values
Correctly Classified Instances	118 (59%)
Incorrectly Classified Instances	82 (41%)
Overall Accuracy	38.14%
Total Number of Instances	200
Time taken to build model	0.03 seconds

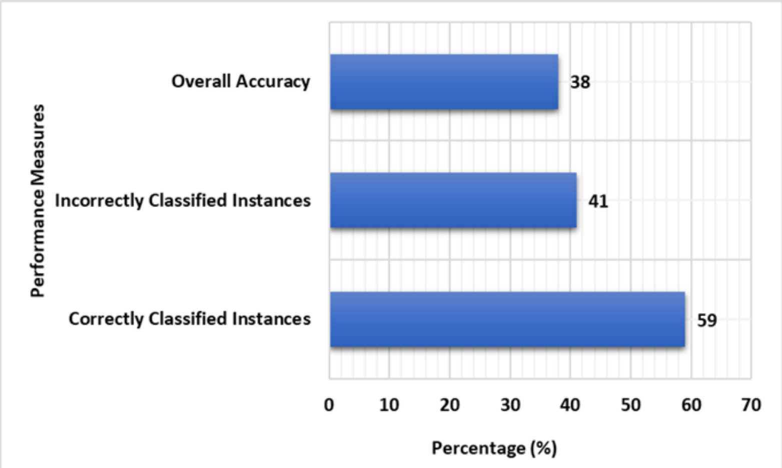

Figure 4.3: Overall Accuracy of Hierarchical Clustering

Table 4.4: Hierarchical Clustering Node or Class wise Results

S. No.	n (truth)	n (classified)	Accuracy	Precision	Recall	F1 Score	Class
1	150	74	0.58	0.97	0.48	0.64	Node1
2	4	40	0.78	0.025	0.25	0.045	Node2
3	39	40	0.59	0.00	0.00	0.00	Node3
4	1	40	0.79	0.025	1	0.049	Node4

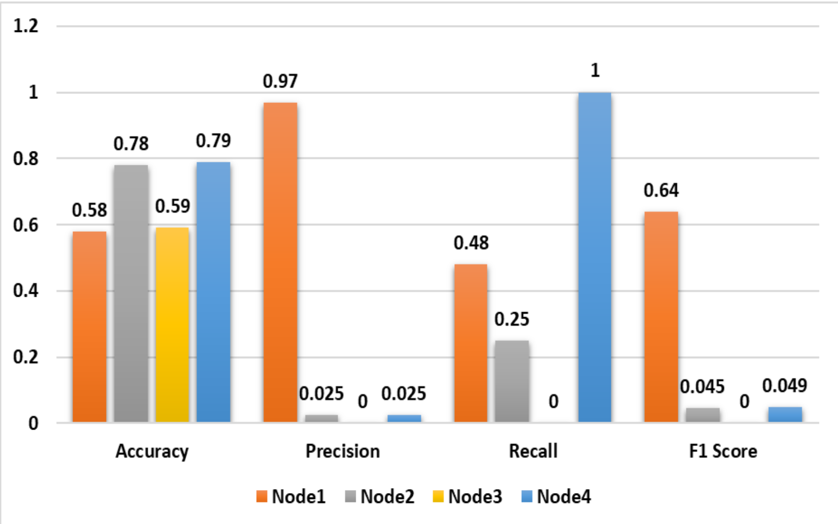

Figure 4.4: Hierarchical Clustering Node or Class wise Results

Based on performance measure it can be concluded that the number of correctly classified instances were 118 in Hierarchical Clustering which confirms overall accuracy of 59%, the incorrectly classified instances were 82. The overall accuracy value is quite less as compared to Canopy Clustering which suggest that Hierarchical Clustering algorithm performance is very poor as compared to Canopy Clustering. Also, the precision & F-measure values of 0.025

and 0.049 respectively were quite lower in comparison to Canopy Clustering.

Make Density Based Clustering:

The Make Density Based Clustering algorithm was being analysed for task scheduling and resource allocation in Fog environment. Accordingly, the results are shown below in the table and figure.

Table 4.5: Overall Accuracy of Make Density Based Clustering

Measures	Values
Correctly Classified Instances	97 (48.5%)
Incorrectly Classified Instances	103 (51.5%)
Overall Accuracy	19.5%
Total Number of Instances	200
Time taken to build model	0.01 seconds

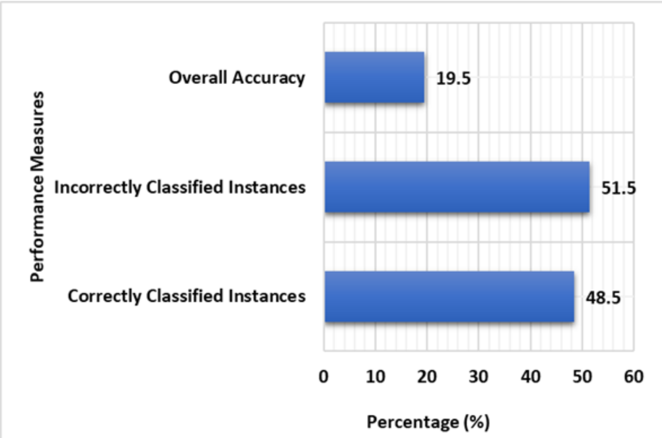

Figure 4.5: Overall Accuracy of Make Density Based Clustering

Table 4.6: Make Density Based Clustering Performance Node wise

S. No.	n (truth)	n (classified)	Accuracy	Precision	Recall	F1 Score	Class
1	50	80	0.61	0.33	0.52	0.40	Node1
2	40	40	0.60	0.00	0.00	0.00	Node2
3	55	40	0.65	0.33	0.24	0.27	Node3
4	55	40	0.52	0.00	0	0.00	Node4

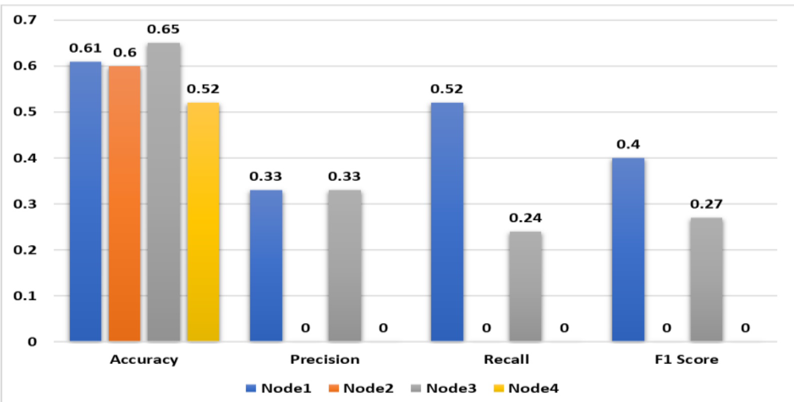

Figure 4.6: Class or Node wise Make Density Based Clustering Performance

Based on the experimental results it can be interpreted that the number of correctly classified instances were 97 in case of Make Density Based Clustering algorithm which is quite lower as compared to other two algorithms Hierarchical Clustering and Canopy Clustering. The overall accuracy value is quite less with value 48.5% among all. Also, it can be concluded that F-measure, Recall and Precision values were lower when compared to Hierarchical Clustering and Canopy Clustering.

For task scheduling and resource allocation the best and most appropriate clustering algorithm is found to be Canopy Clustering.

5. Conclusions:

Finally, the comparative analysis suggest that based on overall accuracy performance measure the most appropriate clustering algorithm identified is Canopy Clustering with highest accuracy value of 74.5%. The task scheduling and resource allocation predictions are better in case of Canopy Clustering as compared to Hierarchical Clustering and Make Density Based Clustering algorithm. The hypothesis testing results concludes that the null hypothesis ($H_0$1) is being rejected which means there is significant difference between various clustering algorithms based on performance measure overall accuracy.

References:

1. E. Ghaffari, “Providing a new scheduling method in fog network using the ant colony algorithm”, Collection of Articles on Computer Science (2019).1 URL https://www.scipedia.com/public/Ghaffari_2019a.
2. Yin, Luxiu& Luo, Juan & Luo, Haibo. (2018). Tasks Scheduling and Resource Allocation in Fog Computing Based on Containers for Smart Manufacture. IEEE Transactions on Industrial Informatics. PP. 1-1. 10.1109/TII.2018.2851241.
3. D. Rahbari and M. Nickray, "Scheduling of fog networks with optimized knapsack by symbiotic organisms search," 2017 21st Conference of Open Innovations Association (FRUCT), 2017, pp. 278-283, doi: 10.23919/FRUCT.2017.8250193.
4. Amit Joshi, Bhavesh Joshi and Manuj Joshi (2012). An approach Initiating Security Protocol towards Cloud Computing. IJCA Proceedings on National Workshop-Cum-Conference on Recent Trends in Mathematics and Computing 2011 RTMC(7).
5. Maitree Desai and Reshma Dayma (2022). Task Scheduling in Fog Environment Using Machine Learning. Journal of Emerging Technologies and Innovative Research (JETIR), Volume 9, Issue 5, pp.-a755-a767.