Annals of R.S.C.B., ISSN:1583-6258, Vol. 24, Issue 2, 2020, Pages. 77 - 92
Received 24 October 2020; Accepted 15 December 2020.

Resource Management in Fog Computing Using Clustering Techniques: A Systematic Study

Swati Malik, Kamali Gupta[1], Malvinder Singh[1]
Chitkara University Institute of Engineering & Technology, ChitkaraUniversity ,Punjab, India
swati.malik@chitkara.edu.in, kamali.singla@chitkara.edu.in, malvinder.singh@chitkara.edu.in

Abstract: As the IoT is evolving day by day in order to provide optimal services to end users, various Emerging technologies of clouds are being integrated with cloud IoT to provide seamless delivery of services to the end users. For end devices it is beneficial to take best out of the emerging technologies of this fog-cloud scenario. In our paper, various resource management approaches used to overcome load on Fog Nodes are discussed. A study of different Clustering based techniques used by different researchers in the area of machine learning is presented. Some of the significant works are highlighted in the literature survey where clustering techniques are implemented on nodes to enhance resource utilization.

Keywords: Clustering, Fog Node, Cloud Computing, resource management.

1.**Introduction to Cloud Computing**

Cloud is the most discussed buzzwords in today's technical world. Cloud in computing terms can be defined as a pool of systems which are interconnected to deliver on demand services to the end users in a virtual environment across the globe. With the advent of cloud to the technological world a new era of computing takes birth. The term "cloud" is a metaphor for the network belonging to any public internet or private network of an organization [1]. "Computing" means the ability to provide computing capabilities online for manipulation and access of storage, applications and resources. Coinciding the two terms: Cloud and Computing, a new technological advancement is ruling the internet world by extending its services to the users magnificently. Cloud has introduced a new concept of everything as a service because it provides services from underlying hardware to the software applications. It is referred as XaaS. In such a system all the components can be measured, delivered and priced accordingly.

Cloud is the network of large set of databases also known as datacenters. A data center is a place dedicated to networked computing devices and storage which provide services like processing and storage of large volume of data[2]. Therefore, cloud computing is the technological advancement in which datacenters are connected with the help of internet in order to provide services to consumers. Cloud does not require local servers or personal computers for managing applications rather it provides centralized resources to be shared between users.

Cloud computing derived its characteristics from some of the established technologies like cluster computing, grid computing and virtualization[3,4]. Cloud computing is now an established paradigm. The paradigm of cloud offers metered capabilities on demand basis to its competing users. Here, on demand means that user needs to pay only for the services of cloud until usage i.e. cost will be dependent on the need only. It implies that the utilization of required services is being done on pay-per-use basis. The term is very much analogous to real-life everyday services utilized by customer such as electricity, internet, water, sewage etc.

2. Emerging Technologiesin Cloud Computing

Generally, cloud computing needs applications which follow a two-tier architecture. In such an architecture one tier is the end user devices which are using the services of cloud these are also called front end nodes. As we all Know nowadays utilization of sensors and smart devices (like smartphones, wearables) is increasing day by day which leads to generation of large volumes of data. This data needs to be managed by the cloud. So, all the techniques related to its maintenance and extraction of useful

Annals of R.S.C.B., ISSN:1583-6258, Vol. 24, Issue 2, 2020, Pages. 77 - 92
Received 24 October 2020; Accepted 15 December 2020.

information from this data has to managed by database logic and business skills[5,6]. This scenario will result in several challenges for computing and also it will be difficult to maintain Quality of service with the conventional infrastructure and resources. Some newer techniques are needed in order to cope up with these demands. One possibility is to create centralized cloud data centers to maintain this large volume of data but this approach will not resolve the problem. One different approach is to extend the computing systems, resources and infrastructure towards the end users which was not supported by traditional cloud computing. However, architecture of cloud computing is improving day by day and accommodating the new technologies. Some of new computing models are include fog computing , serverless computing, mobile edge computing, volunteer computing and mobile edge computing [6].

2 .1 Volunteer computing

This is a type of cloud computing in which volunteers provide computing resources closer to user device. We can also call it donation computing [7]. Here volunteers are general public people who have free resources and wants to contribute any project by giving there computing resources like storage and processing capability. Availability in volunteer cloud computing is not necessary [5]. It is a public funded approach for preserving computing, power and storage. However, it has challenges related to security, privacy and availability.

2.2 Fog and edge computing

The fog computing makes computing resources available near the edge nodes. These edge nodes could be base stations, switches, routers or any other additional computing capability [5]. These fog nodes or edge nodes have limited computability [4]. There are several advantages of fog computing over cloud computing. They have minimized latency time and better quality of service and experience. Fog computing will help in implementing internet of things.

2.3 Serverless computing

Conventional computing using cloud scenario supports mobile applications on a virtual machine which further offer their services to the clients. The cost calculation of such a scenario is based on per VM perhourand this cost also includes the idle time i.e. when VMs were free and not doing any useful task. But if data enter is decentralized then they will consume less power as compared to conventional system. Therefore, it will not be beneficial to pay for server when they are not providing productivity [6].In such cases fog computing will be idle as it is having the capability to mold itself according to requirement. In this cost will depend upon the time it was utilized and number of requests it has processed. Serverless computing is not the computing without server rather it provides an environment in which cost does not include the idle time of server [7]. There are several challenges in implementing this technique as server cannot be idle according to currently used applications. So, applications need to be redesigned for serverless computing which is a tedious task.

2.4 Software-defined computing

All over the world large volumes of data is produced by internet. One of the reasons is number of devices is increasing rapidly. As a result, traffic over the internet is also increasing at a faster pace. This large volume of data belonging to different applications can be shifted or transferred from one cloud to another. In order to support this increasing demand of computing and services a special mechanism is required to cope up with this demand [7].A dynamic architecture is required to support such a system. A new concept is needed which can separate the network infrastructure from the components that manages and control the traffic, this is known a software defined computing. However, there are several challenges to implement this technique. This requires storage or cloud to be physically distributed and logic to manage and control to be centralized.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 24, Issue 2, 2020, Pages. 77 - 92
Received 24 October 2020; Accepted 15 December 2020.

3. Introduction to fog Computing and Resource Management

The aim of the Internet of Things (IoT) is to connect all devices (e.g. smart cameras, Wearables, monitoring devices, smart household appliances, and automobiles) to the Internet, resulting in large data volumes that can overload storage systems and applications for data analytics [8]. There are many applications which require security of critical data, low latency IoT data processing and transmission, example of such applications are health monitoring and early warning systems. The Fog computing model has been suggested to address this constraint, where cloud services are pushed to the edge devices to minimize latency and congestion over the network, as well as to facilitate improved user and end device mobility, and site aware computing.

To understand the full scope for real-time analytics of the Fog and IoT paradigms, several issues ought to be addressed and resolved, such as multi-tenancy, fog aggregation, clustering of fog nodes and delivery of fog facilities[9]. One crucial challenge is the implementation of resource management strategies to decide which analytics platform modules are moved to which edge devices to optimize the usage of fog resources and reduce causes of delays while fulfilling the application's QoS specifications.Resource management usually deals with allocation and deallocation of computing and storage resources. In addition to this, choosing a node or a group of nodes as the suitable match for resource allocation is an important feature of fog resource management. Recent fog network implementations, such as smart city networks, are also implemented on shared fog platforms simultaneously. Therefore, in progress, maximizing the distribution of capital becomes more important.

3.1 Fog Computing Overview

Data Generated via Sensor devices and actuators is actually transferred to distant cloud servers for Storage and Processing purposes. The architecturewill not be accepted in the coming time. It is a de-Facto model for most of the Internet based applications nowadays as this will increase network traffic leading to congestion and Network Latencies. This will degrade the performance of the entire Communication network[10]. An alternative solution to this problem is use of Fog computing where we can bring all the computing resources closer to the nodes i.e Sensor nodes and Actuators for processing data

Fog Computing model will help to reduce the overall traffic which effects the Network bandwidth creating congestion and Latency issues. Distributed services will be provided much closer to the proximity of end user. Current research is undergoing to develop algorithms-based models to decentralize computing resources to bring data closer to the edge node as shown in Fig 1. Different vendors are manufacturing devices which are equipped to provide computing capabilities at the edge of the nodes.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 24, Issue 2, 2020, Pages. 77 - 92
Received 24 October 2020; Accepted 15 December 2020.

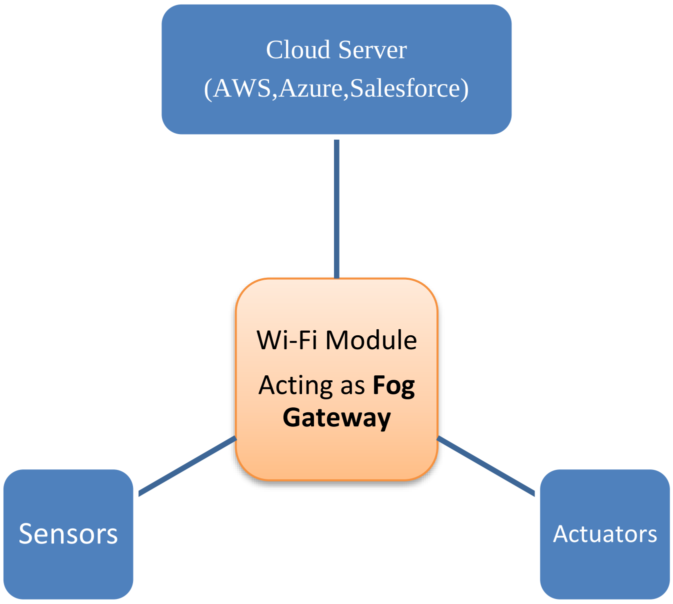

Figure 1. The Model of IoT

3.2 Architecture based approaches for Managing resources in Fog Computing

• **Architecture based on flow of Data**: In this Architecture, we can move data from centralized server to the edge node which will help to distribute the load, thus improving the Latency of the data.
• **Architectures based on the Control**: Here we can deploy single handed control by the centralized server or distributed control by different systems to control or manage the resources.

3.3 Resource Scheduling approach to manage resources in Fog Computing

In Scheduling, we search an optimal solution from a given solution space for N no of Tasks keeping various QoS parameters like Cost, deadline etc into consideration on different fog nodes.

Resource scheduling includes three main approaches i.e. Static scheduling, Dynamic scheduling, and hybrid Scheduling. In the static scheduling approaches [8–11], before submitting task, first we should make prior scheduling before submitting tasks concurrently at fog nodes. This approach is not practically possible as it is not feasible to have prior knowledge of all the resources due to heterogeneity of Fog nodes and Cloud Computing. In the dynamic scheduling approaches [8–18], different tasks are scheduled as per their arrival in the system as arrival time of tasks is not known beforehand. The hybrid approach combines both the concepts to cover scheduling via both static and Dynamic.

3.4Task offloading approach

The purpose of Task offloading is to balance the load between nodes, manage data and Latency, security, to improve energy efficiency and so on.Task offloading concept transfers load from device with limited computational capability (offloading source) to the device having high computational features (offloading destination) to enhance the performance of the IoT based Fog computing model. The task offloading can be categorized into two groups i.e. single type offloading and multi-type offloading. In the first approach [19, 20–24, 27, 28, 30, 34, 35, 38], task with high computation can be offloaded to one single fog node where sequential processing is done. In the multiple-type offloading approaches [17, 18, 25, 26, 27, 31,

Annals of R.S.C.B., ISSN:1583-6258, Vol. 24, Issue 2, 2020, Pages. 77 - 92
Received 24 October 2020; Accepted 15 December 2020.

32, 33, 36, 37], we can offload the data at more than one destination to achieve efficiency (like low latency response) using parallel processing.

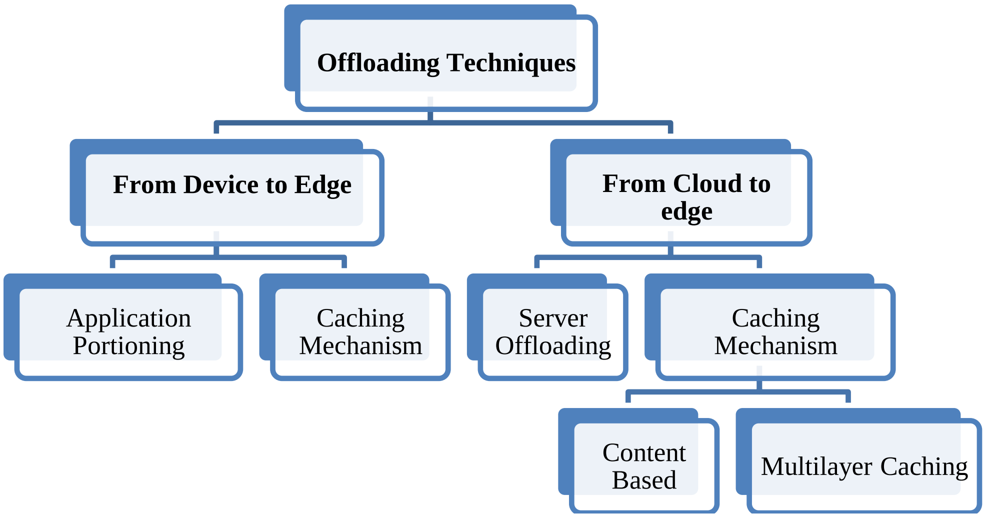

Figure 3: Offloading Techniques

Offloading via User Device to Edge: Here we use twotechniques namely applicationpartitioning and caching mechanisms which are discussed briefly in subsections below.
[37,38,39,40]
a. Application Partitioning: One example of offloading from devices to the edge via applicationpartitioning is in theGigaSight architecture in which Cloudlet VMs [41] are used to process videosstreamed from multiple mobile devices [42].

b**. Caching Mechanisms**: Here a Cache is made globally available to each edge node which will act like a shared memory to different devices who want to interact.

Offloading via Cloud to the Edge: Here, we transfer task from cloud to the edge, the two important techniques used to do this include Server offloading and Caching Technique [43].

a. Server Offloading: In this approach, the workload of the server that resides on the cloud is done either using two ways i.e. replication of data or by portioning. This method is not a good approach to replicate cloud server data on the edge node.

b. Caching Mechanisms: Two techniques under caching approach are discussed below:

Content Popularity–based: Two techniques used to avoid network congestion under this approach include Network based Content-Delivery Networks and cachingbased ISP. But these approaches also suffer a major challenge like exponential growth of IoT devices. To overcome all this, research is undergoing like caching of proactive data at base stations using Hadoop cluster.

Multi-layer Caching: This technique is used to deliver content to wireless based sensor networks [44].

3.5 Load Balancing approach

For applications having Latency issues, balancing the workload at different Fog Nodes is the major issue in IoT based Fog computing. To minimize the response time and enhance Throughput, task sharing

Annals of R.S.C.B., ISSN:1583-6258, Vol. 24, Issue 2, 2020, Pages. 77 - 92
Received 24 October 2020; Accepted 15 December 2020.

between different distributed Fog will help to overcome under Load or Over Load for Fog nodes.Load Balancing faces many challenges in Fog Based Computing like Latency issues in the Network, low priority-based tasks will have to wait in the Queue, and Continuous migration of the Processes will reduce system performance and lack of proper standard for various scenarios[45]. Cloud based Load balancing strategies cannot be applied on Fog Node due to heterogeneity of Fog. Figure 2 depicts the load balancing scenario in a fog-based Model.

As per the Literature survey, there are three approaches to handle Load Balancing: centralized, decentralized, and hybrid. Centralized architecture [46, 47, 48, 45, 50, 51, 52], performs load balancing with the help of central node. This node provides all the tasks related to load balancing. In decentralized architecture [49, 53, 54, 55, 56, 58, 59] all the nodes in the system are organized into different clusters with each cluster consisting of two or more nodes and each cluster node uses central node to perform load balancing. Decentralized architectures are much used in managing resources due to its decentralized control. So, hybrid approach [46, 57] provides better solution then centralized and decentralized architecture.

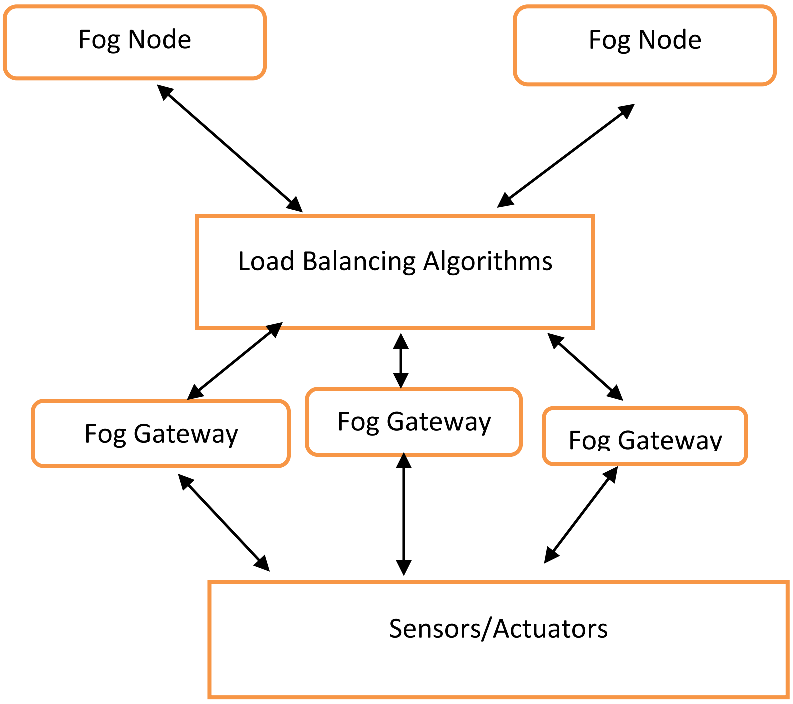

Figure 2: Load Balancing Approach

3.6 Resource Provisioning approach

The Provisioning of resources also called as Auto-Scaling approach is categorized into three scenarios based on time: reactive, proactive, and hybrid policies [60, 61, 62 and 63]. The first one called reactive policy responds to current system status without any Prediction Techniques. In Reactive approach, applications [64, 65–67, 72, 76] are scaled down after change in workload has happened. In Proactive approach, [73, 76, 78] it uses techniques such as Neural networks, Time series etc to predict future demands in IoT based applications. In hybrid approach [74, 75], combined reactive policy to scale out (inclusion of a new fog node) & Proactive policy to scale in (i.e. to release a fog node) decisions are taken [79, 80].

Annals of R.S.C.B., ISSN:1583-6258, Vol. 24, Issue 2, 2020, Pages. 77 - 92
Received 24 October 2020; Accepted 15 December 2020.

There are different approaches for managing resource in fog environment,it requires a need to address the issue of clustering of fog nodes as this plays a vital role inoverall resource management process. If all the nodes are clustered effectively then subsequent steps of assigning the edge devices to the requested process can be done efficiently with minimum delay.Basically, clustering techniques are studied in machine learning. So, in order to have a detailed idea of clustering in the next section clustering and some of its basic techniques are discussed so that it could be understood what are the basic techniques of clustering and how can they can be implemented in fog environment. In the upcoming section a literature a survey is presented in which various authors have done significant work in clustering of fog nodes. A little work has been done in this area still a lot of work can be done so that fog computing can achieve heights in terms of performance.

4 Clustering

Clustering is generally associated with machine learning in order to classify data in various clusters. In this section a general overview of clustering and its types is provided. Later in upcoming section clustering techniques implemented in fog computing will be discussed. Clustering can be defined as organization of unclassified data into similarity groups called “clusters”. As data over the data centers is unclassified and useful information such as trends and patterns in data can be discovered only when it is classified in a systematic manned. A cluster is a collection of data items which are “similar” between them, and “dissimilar” to data items in other clusters [81]. Clustering has various applications in real life scenarios such as marketing, biology, insurance, city planning, earthquake studies etc. Some of the reasons why we should consider clustering an important task are: It can be expensive to mark a wide collection of sample patterns. There could be no knowledge of the contents of the database. For identifying functionality that would later be useful for categorization, clustering may be used. It will help to gain insight into the data's existence [82]. It may lead to the detection of different subclasses or pattern similarities. Figure 3 shows the types of clustering technique.

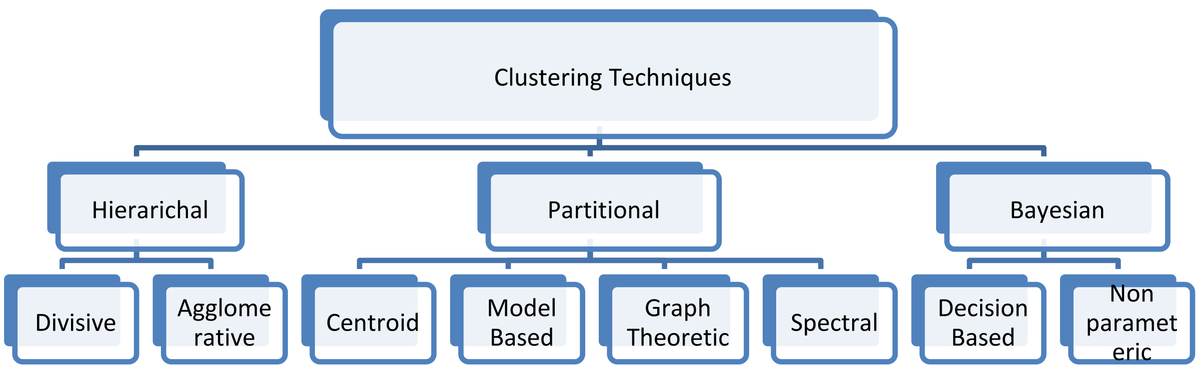

Figure 3: Clustering Techniques

Hierarchical technique finds progressive clusters utilizing recently created clusters. These calculations could be either agglomerative or divisive. Agglomerative hierarchal technique called "bottom up" approach and divisive as "top down" approach.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 24, Issue 2, 2020, Pages. 77 - 92
Received 24 October 2020; Accepted 15 December 2020.

Divisive strategy starts with entire cluster and continues to isolate it into progressively more modest cluster [83]. It begins with all information focuses in one cluster, the root. After that root cluster is being split into smaller clusters. Every newly formed cluster is recursively partitioned further. This recursive partitioningstops when just singleton cluster of individual information is left.

Agglomerative procedure starts with every component as a different cluster and consolidations them into progressively bigger clusters [84]. Agglomerative (base up) grouping the dendrogram is worked from the base level by two different ways. Firstly by combining the most comparable (or closest) pair of clusters and furthermore halting when all the information focuses are converged into a solitary cluster (i.e., the root group).

Partitional Techniquesfollows the approach of finding all clusters in one go. Partitional technique can be categorized as divisive hierarchical clustering. Partitonal clustering technique can be further categorized as centeroid, model based, and graph theoretic and spectral. Centroid clustering technique refers to the k-means clustering[85]. K-means was introduced by MacQueen in 1967. This algorithm divides the given data set into k clusters. Every cluster in the data set has a center, this center is known as centroid k.

K-means Simplified Procedure is enlisted step by step as follows:

Step1: Initially find “k” centroid by selecting random data points from available data set, these centroids could be called as cluster centers.

Step2: Each data point is assigned to the nearest centroid available.

Step 3: Centroids need to be computed again by utilizing most recent cluster membership data points.

Step 4. Steps 2 and 3 need to be repeated unless terminating criteria is met with. Sometimes terminating citeria is called as convergence criteria.

Strengths of K- mean algorithm is easy to understand and implement. Time complexity of this algorithm is given as O(nkt), here n, k and t refers to different data points, different clusters, and number of repetitions respectively[86]. This algorithm is almost linear as number of clusters and time are very small values. K-means is the most mainstream clustering algorithm utilized by various applications. However, this algorithm stops at a local optimal solution if sum of squared error is utilizsed. The global optimal solution is difficult to trace because of the underlying complexity of the algorithm.

Despite of several advantages and strength K-means algorithm has some weaknesses as well. This algorithm is appropriate only in case the mean is provided beforehand. For definitely categorized data, the centroid is represented by most frequent data points in that data set. One other problem with this algorithm is that k needs to be specified by user [87]. Susceptibility towards outliers is an important concern in K- means algorithm. Those data points that are at large distance or far away in value of parameters from other data points are known as outliers. Outliers could be errors and anomalies in the information recording or some some extraordinary informationwith altogether different qualities.

Model-based clustering is based upon the assumption that the data is produced by a model and by utilizing this information the first model can be recuperated. The model that is recuperated from the data at that point characterizes clusters and an assignment of documents to clusters.Model-based clustering gives a structure to join information for creating cluster. Informational indexes provide a framework for incorporating knowledge or data sets about a domain. k-means and the hierarchical algorithms make genuinely inflexible presumptions about the data. One of the examples of such presumption is that K-

Annals of R.S.C.B., ISSN:1583-6258, Vol. 24, Issue 2, 2020, Pages. 77 - 92
Received 24 October 2020; Accepted 15 December 2020.

means algorithm presumes that clusters are spherical in shape [88]. Model-based clustering provides additional advantage of adaptability and flexibility. The clustering model can be adapted to what is known about the circulation or characterization of the data. This distribution of data could be Bernoulli or Gaussian with non-spherical variance Expectation Maximization algorithm or EM algorithm is prominently utilized algorithm for model based clustering.

Graph Based Clustering utilizes the nearness data set values in order to make a graphical representation of it. It begins with the formation of proximity matrix. Each data point in data set is treated as a node in a graph. Nodes are connected by edges and each has a weight value which is the nearness or distance between the two data points. At the start the graph is fully linked. Minimum is known as single link and maximum is known as complete link. In the simplest scenario, clusters are considered as connected components in the graph. Execution of Clustering can be improved by utilizing sparsification technique. In this technique nearest neighbor connections are kept as it is where as associations between least similar or less comparable nodes are broken [89]. The closest neighbors of a data point tend to belong to the same class as the point itself. This strategy lessens the effect of noise, outliers and other anomalies encountered during data collection. It improves the differentiation between clusters. Sparsification encourages the use of graph partitioning algorithms. Chameleon and Hypergraph-based Clustering are examples of such algorithms.

Spectral clustering is a procedure inspired from graph theory. In clustering this approach is used to distinguish commonalities of nodes in a graph based on the edges connecting them. The strategy is adaptable and permits us to cluster non graphical data also. Spectral clustering utilizes data from the eigenvalues (spectrum) of uncommon matrices built from the data set.

Bayseian techniques is opposite of agglomerative clustering as k means algorithm returns only one clustering solution. Bayesian technique is nonparametric technique which gives a rear solution over the entire space of divisions. It permits one to assess statistical properties, such as uncertainty on the number of clusters. Decision based clustering is based on a supervised learning technique called as decision tree. The core idea is to utilize decision tree technique to segment the data set into cluster or regions and empty regions,here empty region refer to outliers and anomalies in the data set [90].This can be accomplished by bringing virtual data points into the space and afterwards applying decision tree algorithm. The method can discover clusters in enormous high dimensional spaces productively. This technique is appropriate for clustering in the full dimensional space just as in subspaces.

Nonparametric Bayesian technique givesentirely adaptable modelsin a Bayesian framework. Bayesian techniques are most exact when the earlier techniques sufficiently epitomize one's convictions, nonparametric priors can represent data better than rigid models with the quantity of parameters set before (for example a combination of 3 Gaussians). Numerous nonparametric Bayesian models can be inferred by beginning with a standard parametric model and accepting the breaking point as the quantity of boundaries goes to vastness (for example an endless combination of Gaussians) [91][92].These nonparametric models will naturally induce the right model size (for example number of significant boundaries) from the information, without having to unequivocally perform model correlations (for example contrasting a combination of 3 Gaussians with a combination of 4 Gaussians to decide the right number).

Annals of R.S.C.B., ISSN:1583-6258, Vol. 24, Issue 2, 2020, Pages. 77 - 92
Received 24 October 2020; Accepted 15 December 2020.

Literature Survey:

Table1: Literature Survey:

Year	Title	Parameter	Description	Heuristic used
2012	Combination of Fuzzy C-Means and Particle Swarm Optimization for Text Document Clustering [93]	• Purity values of three algorithms on two different data sets	• hybrid approach introduced based on PSO-FCM • PSO-FCM assists Fuzzy C Means to integrate the jump from the local optima and overcome the slow-moving integration speed of the PSO algorithm.	• a hybrid method of text document clustering based on fuzzy c-means and particle swarm optimization (PSO-FCM)
2018	Dynamic IoT Device Clustering and Energy Management With Hybrid NOMA Systems [94]	• spectrum efficiency • fairness among IoT devices • Total Bandwidth • Downlink Total power budget • Noise power spectrum density • Cell Radius • Carier Frequency • Path Loss • distance between IOT device and Base station	• Integration of downtime IoT devices that do not reach is resolved to reduce system complexity and delays for IoT devices with better channel conditions. • Distributed power management is resolved using a Nash bargaining solution in each set to ensure balance between IoT devices. • Ensures justice between IoTs compared to other schemes.	• hybrid nonorthogonal multiple access (OR) to provide communication services between the fog layer and (IoT) device layer in the fog computing, • Nash Bargaining Solution (NBS) • Nonorthogonal multiple access • (OR), • Strong Collaboration Framework (meeting) • Link to NOMA
2019	Balanced clustering and joint resources allocation in cooperative fog computing system [95]	• computational delay • energy consumption	• Aggregated and limited resource allocation resources (BCJRA) to achieve applied response delays and co-operative use between adjacent fog areas.	• • Algorithm for shared and equitable resource collection algorithm • The hardware integration algorithm generates clusters based on the distance between fog nodes, wireless resources and resources.
2019	Methods of Resource Scheduling based on optimized Fuzzy Clustering	• Task requests types and resource type : computing resource, storage resource	• FCAP algo=PSO+FCM • FCM decide quantity to which each	• Fuzzy C(clustering)-means algo • Particle swarm optimization

	in Fog Computing [96]	• Clustering accuracy rate • Objective function value(convergenc e speed) • User satisfaction index (for evaluating rationality of scheduling)	sample point belong to each associated cluster • Membership degree value lies interval[0,1] • FCAP determine cluster center and represent cluster center with one particle in PSO • Fitness level is calculated, based on that global and local position are calculated • Comparison of RSAF algo with min minalgo on the basis of user satisfaction	• RSAF algo for scheduling
2020	Designing an efficient clustering strategy for combined fog to cloud Scenerio [97]	• Average FCC cost • Average solving time	• FCC problem formulated as MILP (mixed integer Linear Programming) • Minimum and maximum liability is obtained for the required number of collections • Enhance the durability of architectural art with a backup device in the collection. • A machine-based learning device that provides measurable and closest solutions to real-world solutions has been proposed where, due to the high number of connected devices, MILP formatting does not work.	Unsupervised machine learning K-means based heuristic

Conclusion and Future Work

Fog computing is the most sorted out emerging technology out of all technologies which are evolved from cloud. Generally, Fog computing is integrated with Internet of Things which can provide better services as t is near to end devices. It can provide services with minimum delay. Clustering of fog nodes plays a vital role in scheduling fog nodes for fog to cloud scenario. As scheduling of tasks and resources can be done efficiently only when available devices are clustered properly. Some of the researchers have implemented various clustering algorithms. A lot of work can be done in improving clustering techniques and algorithms in computing. Dynamic and heterogeneous nature of fog nodes should be taken into account so that better customer satisfaction and efficient utilization of resources can be achieved.

References

1. Rajkumar Buya, Christian Veccilo, S. ThanaraiSelvi, Mastering Cloud Computing, Rajkumar Buya, Christian Veccilo, S. ThanaraiSelvi, Mc Graw Hill Education 2018
2. Thomas Erl, Ziagham Mahmood and Ricardo Puttini, Cloud Computing Concepts, Technology and Architecture 2017
3. https://www.techwell.com/techwell-insights/2017/10/6-major-challenges-cloud-computing
4. https://www.geeksforgeeks.org/cloud-computing-research-challenges/
5. FredericoDurao · Jose Fernando S. Carvalho et al. A systematic review on cloud computing. Springer Science+Business Media New York 2014
6. Blesson Varghese, Rajkumar Buyya.Next generation cloud computing: New trends and research directions Future Generation Computer Systems 79 (2018) 849–861.
7. F. Costa, L. Silva, M. Dahlin, Volunteer cloud computing: Mapreduce over the internet, in: IEEE International Symposium on Parallel and Distributed Processing Workshops, 2011, pp. 1855–1862
8. Sun, Y., Lin, F., Xu, H.: Multi-objective optimization of resource scheduling in fog computing using an improved NSGA-II. Wirel. Pers. Commun. 102(2), 1369–1385(2018)
9. Bitam, S., Zeadally, S., Mellouk, A.: Fog computing job scheduling optimization based on bee’s swarm. Enterprise Information Systems (EIS). 12(4), 373–397 (2017)
10. Cardellini, V., et al. On QoS-aware scheduling of data stream applications over fog computing infrastructures. In Computers and Communication (ISCC), 2015 IEEE Symposium on. IEEE (2015)
11. De Benedetti,M., et al.: JarvSis: a distributed scheduler forIoT applications. Clust. Comput. 20(2), 1775–1790 (2017)
12. Zeng, D., Gu, L., Guo, S., Cheng, Z., Yu, S.: Joint optimization of task scheduling and image placement in fog computing supported software-defined embedded system. IEEE Trans. Comput. 65(12), 3702–3712 (2016)
13. Fan, J., et al. Deadline-Aware Task Scheduling in a Tiered IoT Infrastructure. in GLOBECOM 2017–2017 IEEE Global Communications Conference. Singapore: IEEE (2017)
14. Rahbari, D. and M. Nickray. Scheduling of Fog Networks with Optimized Knapsack by Symbiotic Organisms Search. In 2017 21st Conference of Open Innovations Association(FRUCT). Finland: IEEE (2017)
15. Pham, X.-Q. and E.-N. Huh. Towards task scheduling in a cloud-fog computing system. In Network Operations and Management Symposium (APNOMS), 2016 18th Asia- Pacific. IEEE (2016)
16. Kabirzadeh, S., D. Rahbari, and M. Nickray, A Hyper Heuristic Algorithm for Scheduling of Fog Networks. algorithms.19: p. 20 (2017)
17. Sun, Y., Zhang, N.: A resource-sharing model based on a repeated game in fog computing. Saudi journal of biological sciences (SJBS). 24(3), 687–694 (2017)
18. Hoang, D. and T.D. Dang, FBRC: Optimization of taskScheduling in Fog-Based Region and Cloud. 2017: p.1109–1114.
19. Chen, X.,Wang, L.: Exploring fog computing-based adaptive vehicular data scheduling policies through a compositional formal method—PEPA. IEEE Commun. Lett. 21(4), 745–748 (2017)
20. Urgaonkar, R., Wang, S., He, T., Zafer, M., Chan, K., Leung, K.K.: Dynamic service migration and workload scheduling in edge-clouds. Perform. Eval. 91, 205–228(2015)
21. Bittencourt, L.F., Diaz-Montes, J., Buyya, R., Rana, O.F., Parashar,M.: Mobility-aware application scheduling in fogcomputing. IEEE Technical Committee on Cloud Computing (TCCLD). 4(2), 26–35 (2017)

22. Deng, R., et al.: Optimal workload allocation in fog-cloudcomputing towards balanced delay and power consumption.IEEE Internet Things J. 3(6), 1171–1181 (2016)
23. Tran, D.H., Tran, N.H., Pham, C., Kazmi, S.M.A., Huh,E.N., Hong, C.S.: OaaS: offload as a service in fog networks.Computing. 99(11), 1081–1104 (2017)
24. Mukherjee, A., Deb, P., de, D., Buyya, R.: C2OF2N: a lowpower cooperative code offloading method for femtoletbasedfog network. J. Supercomput. 74(6), 2412–2448(2018)
25. Liu, L., Chang, Z., Guo, X., Mao, S., Ristaniemi, T.:Multiobjective optimization for computation offloading in fog computing. IEEE Internet Things J. 5(1), 283–294(2018)
26. Wang, X., Ning, Z., Wang, L.: Offloading in internet ofvehicles: a fog-enabled real-time traffic management system.IEEE Trans. Ind. Inf. 14(10), 4568–4578 (2018)
27. Liu, L., Z. Chang, and X. Guo, Socially-aware DynamicComputation Offloading Scheme for Fog Computing System with Energy Harvesting Devices. IEEE InternetThings J.. p. 1–1 (2018)
28. Xu, J. and S. Ren. Online learning for offloading andautoscaling in renewable-powered mobile edge computing. In Global Communications Conference (GLOBECOM),2016 IEEE. IEEE (2016)
29. Zhao, X., L. Zhao, and K. Liang. An Energy ConsumptionOriented Offloading Algorithm for Fog Computing. InInternational Conference on Heterogeneous Networkingfor Quality, Reliability, Security and Robustness. Springer (2016)
30. Ye, D., et al., Scalable Fog Computing with ServiceOffloading in Bus Networks. p. 247–251 (2016)
31. Meng, X., Wang,W., Zhang, Z.: Delay-constrained hybridcomputation offloading with cloud and fog computing IEEE (ACCESS). 5, 21355–21367 (2017)
32. Nan, Y., Li, W., Bao, W., Delicato, F.C., Pires, P.F.,Zomaya, A.Y.: A dynamic tradeoff data processing frameworkfor delay-sensitive applications in cloud of thingssystems. J. Parallel Distrib. Comput. 112, 53–66 (2018)
33. Chamola, V., C.-K. Tham, and G.S. Chalapathi. Latencyaware mobile task assignment and load balancing for edgecloudlets. In Pervasive Computing and CommunicationsWorkshops (PerComWorkshops), 2017 IEEE InternationalConference on. IEEE (2017)
34. Alam, M.G.R., Y.K. Tun, and C.S. Hong. Multi-agent andreinforcement learning based code offloading in mobilefog. In Information Networking (ICOIN), 2016International Conference on. IEEE (2016)
35. Khan, J.A., C. Westphal, and Y. Ghamri-Doudane.Offloading Content with Self-organizing Mobile Fogs. InTeletraffic Congress (ITC 29), 2017 29th International.IEEE (2017)
36. Ahn, S.,M. Gorlatova, andM. Chiang. Leveraging fog andcloud computing for efficient computational offloading. InUndergraduate Research Technology Conference (URTC),2017 IEEE MIT. IEEE (2017)
37. Bozorgchenani, A., D. Tarchi, and G.E. Corazza. AnEnergy-Aware Offloading Clustering Approach (EAOCA) in fog computing. In Wireless Communication Systems(ISWCS), 2017 International Symposium on. IEEE (2017)
38. Zhu, Q., Si, B., Yang, F., Ma, Y.: Task offloading decisionin fog computing system. China Communications(Chinacom). 14(11), 59–68 (2017)
39. Chang, Z., et al. Energy Efficient Optimization forComputation Offloading in Fog Computing System. InGLOBECOM 2017-2017 IEEE Global CommunicationsConference. IEEE (2017)
40. Bozorgchenani, A., D. Tarchi, and G.E. Corazza. AnEnergy and Delay-Efficient Partial Offloading Techniquefor Fog Computing Architectures. In GLOBECOM 2017-2017 IEEE Global Communications Conference. IEEE(2017)
41. Bao,W., et al. Cost-Effective Processing in Fog-IntegratedInternet of Things Ecosystems. In Proceedings of the 20thACMInternational Conference on Modelling, Analysis andSimulation of Wireless and Mobile Systems. ACM (2017)
42. Mahadev Satyanarayanan, Paramvir Bahl, Ramon Caceres, and Nigel Davies. 2009. The case for VM-based cloudletsin mobile computing. *IEEE Pervas. Comput.* 8, 4 (2009).
43. Pieter Simoens, Yu Xiao, Padmanabhan Pillai, Zhuo Chen, Kiryong Ha, and Mahadev Satyanarayanan. 2013. Scalablecrowd-sourcing of video from mobile devices. In *Proceedings of the 11th International Conference on Mobile Systems,Applications, and Services*. ACM, 139–152.
44. Engin Zeydan, Ejder Bastug, Mehdi Bennis, Manhal Abdel Kader, Ilyas Alper Karatepe, Ahmet Salih Er, and Merouane Debbah. 2016. Big data caching for networking: Moving from cloud to edge. *IEEE Commun. Mag.* 54,9 (2016), 36–42.
45. Thang X. Vu, Symeon Chatzinotas, and B. Ottersten. 2017. Energy-efficient design for edge-caching wireless networks:When is coded-caching beneficial? In *Proceedings of the IEEE 18th InternationalWorkshop on Signal ProcessingAdvances in Wireless Communications (SPAWC'17)*. 1–5. DOI:10.1109/SPAWC.2017.8227689

Annals of R.S.C.B., ISSN:1583-6258, Vol. 24, Issue 2, 2020, Pages. 77 - 92
Received 24 October 2020; Accepted 15 December 2020.

46. Liang, K., Zhao, L., Zhao, X., Wang, Y., Ou, S.: Jointresource allocation and coordinated computationoffloading for fog radio access networks. ChinaCommunications (Chinacom). 13(2), 131–139 (2016)
47. Perala, S.S.N., I. Galanis, and I. Anagnostopoulos. FogComputing and Efficient Resource Management in the era of Internet-of-Video Things (IoVT). In Circuits andSystems (ISCAS), 2018 IEEE International Symposiumon. IEEE (2018)
48. Chen, X., Jiao, L., Li, W., Fu, X.: Efficient multi-usercomputation offloading for Mobile-edge cloud computing. IEEE/ACM Trans. Networking. 24(5), 2795–2808 (2016)
49. Kattepur, A., et al. Resource constrained offloading in fogcomputing. In Proceedings of the 1st Workshop on Middleware for Edge Clouds & Cloudlets. ACM (2016)
50. Xiong, Z., et al.: Cloud/fog computing resource managementand pricing for blockchain networks. IEEE Internet Things J. 6(3), 4585–4600 (2018)
51. Li, C., Zhuang, H., Wang, Q., Zhou, X.: SSLB: selfsimilarity-based load balancing for large-scale fog computing. Arab. J. Sci. Eng. 43(12), 7487–7498 (2018)
52. Manasrah, A.M., A.a. Aldomi, and B.B. Gupta, An optimizedservice broker routing policy based on differential evolution algorithm in fog/cloud environment. ClusterComputing, (2017)
53. Beraldi, R., A. Mtibaa, and H.Alnuweiri. Cooperative loadbalancing scheme for edge computing resources. In Fogand Mobile Edge Computing (FMEC), 2017 SecondInternational Conference on. IEEE (2017)
54. Shi, C., Z. Ren, and X. He, Research on Load Balancingfor Software Defined Cloud-Fog Network in Real-Time Mobile Face Recognition. 210: p. 121–131 (2018)
55. He, X., Ren, Z., Shi, C., Fang, J.: A novel load balancingstrategy of software-defined cloud/fog networking in the internet of vehicles. China Communications (Chinacom).13(2), 140–149 (2016)
56. Ningning, S., Chao, G., Xingshuo, A., Qiang, Z.: Fogcomputing dynamic load balancing mechanism based on graph repartitioning. China Communications (Chinacom).13(3), 156–164 (2016)
57. Yu, Y., X. Li, and C. Qian. SDLB: A Scalable and DynamicSoftware Load Balancer for Fog and Mobile Edge Computing. In Proceedings of the Workshop on MobileEdge Communications. ACM (2017)
58. Oueis, J., E.C. Strinati, and S. Barbarossa. The fogbalancing: Load distribution for small cell cloud computing. In Vehicular Technology Conference (VTC Spring),2015 IEEE 81st. IEEE (2015).
59. Neto, E.C.P., G. Callou, and F. Aires. An algorithm tooptimise the load distribution of fog environments. InSystems, Man, and Cybernetics (SMC), 2017 IEEEInternational Conference on. . IEEE (2017).
60. Kapsalis, A., Kasnesis, P., Venieris, I.S., Kaklamani, D.I.,Patrikakis, C.Z.: A cooperative fog approach for effectiveworkload balancing. IEEE Cloud Computing. 4(2), 36–45(2017)
61. Verma, S., et al. An efficient data replication and loadbalancing technique for fog computing environment. In Computing for Sustainable Global Development(INDIACom), 2016 3rd International Conference on.IEEE (2016)
62. Gu, L., Zeng, D., Guo, S., Barnawi, A., Xiang, Y.: Costefficient resource management in fog computing supportedmedical cyber-physical system. IEEE Trans. Emerg. Top.Comput. 5(1), 108–119 (2017)
63. Xu, X., Fu, S., Cai, Q., Tian,W., Liu,W., Dou,W., Sun, X.,Liu, A.X.: Dynamic resource allocation for load balancingin fog environment.Wirel. Commun.Mob. Comput. 2018,1–15 (2018)
64. Ni, L., Zhang, J., Jiang, C., Yan, C., Yu, K.: Resourceallocation strategy in fog computing based on priced timed petri nets. IEEE Internet Things J. 4(5), 1216–1228 (2017)
65. Zhang, H., Xiao, Y., Bu, S., Niyato, D., Yu, F.R., Han, Z.:Computing resource allocation in three-tier IoT fog networks:a joint optimization approach combiningStackelberg game and matching. IEEE Internet Things J.4(5), 1204–1215 (2017).
66. Alsaffar, A.A., Pham, H.P., Hong, C.S., Huh, E.N., Aazam,M.: An architecture of IoT service delegation and resourceallocation based on collaboration between fog and cloudcomputing. Mob. Inf. Syst. 2016, 1–15 (2016).
67. Zhang, Y., et al., Resource Allocation in Software DefinedFog Vehicular Networks. 2017: p. 71–76
68. Do, C.T., et al. A proximal algorithm for joint resourceallocation and minimizing carbon footprint ingeo-distributed fog computing. In InformationNetworking (ICOIN), 2015 International Conferenceon. IEEE (2015).
69. Xu, J., et al. Zenith: Utility-aware resource allocation foredge computing. In Edge Computing (EDGE), 2017 IEEEInternational Conference on. IEEE (2017)
70. Aazam, M., et al., IoT resource estimation challenges andmodeling in fog, in Fog Computing in the Internet of Things, Springer. p. 17–31 (2018).
71. Zhang, H., Zhang, Y., Gu, Y., Niyato, D., Han, Z.: Ahierarchical game framework for resource management in fog computing. IEEE Commun. Mag. 55(8), 52–57 (2017)

72. Sood, S.K., Singh, K.D.: SNA based resource optimizationin optical network using fog and cloud computing. Opt.Switch. Netw. 33(July), 114–121 (2017)
73. Kochar, V. and A. Sarkar. Real time resource allocation ona dynamic two-level symbiotic fog architecture. In Embedded Computing and System Design (ISED), 2016Sixth International Symposium on. IEEE (2016).
74. Naranjo, P.G., et al.: Fog over virtualized IoT: new opportunityfor context-aware networked applications and a casestudy. Appl. Sci. 7(12), 1325 (2017)
75. Jiao, Y., et al.: Auctionmechanisms in cloud/fog computing resource allocation for public Blockchain networks. IEEETrans. Parallel Distrib. Syst. 30(9), 1975–1989 (2018)
76. Ali, M., Riaz, N., Ashraf,M.I., Qaisar, S., Naeem,M.: Jointcloudlet selection and latency minimization in fog networks.IEEE Trans. Ind. Inf. 14(9), 4055–4063 (2018)
77. Nguyen, D.T., L.B. Le, and V. Bhargava, Price-basedResource Allocation for Edge Computing: A MarketEquilibrium Approach. arXiv preprint arXiv:1805.02982, (2018)
78. Zhang, W., Zhang, Z., Chao, H.-C.: Cooperative fog computingfor dealing with big data in the internet of vehicles:architecture and hierarchical resource management. IEEECommun. Mag. 55(12), 60–67 (2017).
79. Anglano, C., M. Canonico, andM.Guazzone. Profit-awareresource management for edge computing systems. In Proceedings of the 1st International Workshop on EdgeSystems, Analytics and Networking. ACM (2018)
80. El Kafhali, S., Salah, K.: Efficient and dynamic scaling offog nodes for IoT devices. J. Supercomput. 73(12), 5261–5284 (2017)
81. C. C. Aggarwal and C. K. Reddy (2014). Data Clustering: Algorithms and Applications, Taylor & Francis Group, LLC
82. A. Agresti, “Two Bayesian/frequentist challenges for categorical data analyses,” METRON, vol. 72, no. 2, pp. 125–132, Aug. 2014.
83. P. Jaganathan and R. Kuppuchamy, “A threshold fuzzy entropy based feature selection for medical database classification,” Comput. Biol. Med., vol. 43, no. 12, pp. 2222–2229, 2013.
84. A. Mitchell et al., “The InterPro protein families database: the classification resource after 15 years,” Nucleic Acids Res., vol. 43, no. D1, pp. D213–D221, 2014.
85. I. Pedruzzi et al., “HAMAP in 2013, new developments in the protein family classification and annotation system,” Nucleic Acids Res., vol. 41, no. D1, pp. D584–D589, 2012.
86. A. Oellrich, I. Jacobsen, J. Papatheodorou, M. G. P. Sanger, and D. Smedley, “Using association rule mining to determine promising secondary phenotyping hypotheses,” Bioinformatics, vol. 30, no. 12, pp. i52–i59, 2014
87. Mazure, C. M., and Swendsen, J. (2016). Sex differences in Alzheimer’s disease and other dementias. *Lancet Neurol.* 15, 451–452. doi: 10.1016/S1474-4422(16)00067-3
88. Tosto, G., Monsell, S. E., Hawes, S. E., Bruno, G., and Mayeux, R. (2016). Progression of extrapyramidal signs in Alzheimer’s disease: clinical and neuropathological correlates. *J. Alzheimers Dis.* 49, 1085–1093. doi: 10.3233/JAD-150244
89. Narita K., Hochin T., Hayashi Y., Nomiya H. (2020) Improvement of Incremental Hierarchical Clustering Algorithm by Re-insertion. In: Lee R. (eds) Computational Science/Intelligence and Applied Informatics. CSII 2019. Studies in Computational Intelligence, vol 848. Springer, Cham. https://doi.org/10.1007/978-3-030-25225-0_8
90. Narita, K., Hochin, T., Nomiya, H.: Incremental clustering for hierarchical clustering. In: Proceedings of 5th International Conference on Computational Science/Intelligence and Applied Informatics (CSII 2018), pp. 102–107 (2018). https://doi.org/10.1109/CSII.2018.00025
91. M. P. Naik, H. B. Prajapati and V. K. Dabhi, "A survey on semantic document clustering," *2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, Coimbatore, 2015, pp. 1-10, doi: 10.1109/ICECCT.2015.7226036.
92. M. S. Anbarasi et al., "Ontology Oriented Concept Based Clustering", *IJRET: Int. J. of Research in Eng. and Technology*, vol. 3, no. 2, Feb 2014.
93. Kang J., Zhang W. (2012) Combination of Fuzzy C-Means and Particle Swarm Optimization for Text Document Clustering. In: Xie A., Huang X. (eds) Advances in Electrical Engineering and Automation. Advances in Intelligent and Soft Computing, vol 139. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-27951-5_37

94. X. Shao, C. Yang, D. Chen, N. Zhao and F. R. Yu, "Dynamic IoT Device Clustering and Energy Management With Hybrid NOMA Systems," in IEEE Transactions on Industrial Informatics, vol. 14, no. 10, pp. 4622-4630, Oct. 2018, doi: 10.1109/TII.2018.2856776.
95. H. Cheng, W. Xia, F. Yan and L. Shen, "Balanced Clustering and Joint Resources Allocation in Cooperative Fog Computing System," 2019 IEEE Global Communications Conference (GLOBECOM), Waikoloa, HI, USA, 2019, pp. 1-6, doi: 10.1109/GLOBECOM38437.2019.9013392.
96. Li, G.; Liu, Y.; Wu, J.; Lin, D.; Zhao, S. Methods of Resource Scheduling Based on Optimized Fuzzy Clustering in Fog Computing. *Sensors* **2019**, *19*, 2122
97. Asensio, A. et al. "Designing an efficient clustering strategy for combined Fog-to-Cloud scenarios." *Future Gener. Comput. Syst.* 109 (2020): 392-406.