iJRASET
International Journal For Research in Applied Science and Engineering Technology

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

**Volume:** 11 **Issue:** I **Month of publication:** January 2023

**DOI:** https://doi.org/10.22214/ijraset.2023.48484

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 11 Issue I Jan 2023- Available at www.ijraset.com*

# A Comprehensive Review on Edge Computing

Dr Suresha K[1], Suresh Goure[2], Shaheen Banu[3]
[1, 3]*Department of Computer Science and Engineering, D R R Government Polytechnic Davangere Karnataka India*
[2]*Government Women Polytechnic, Kalburgi, Karnataka, India*

***Abstract:*** ***The Edge computing paradigm has experienced significant growth in both academic and professional circles in recent years. By linking cloud computing resources and services to the end users, it acts as a crucial enabler for several emerging technologies, including 5G, the Internet of Things (IoT), augmented reality, and vehicle-to-vehicle communications. Applications that require low latency, mobility, and location awareness are supported by the edge computing paradigm. Significant research has been done in the field of edge computing, which is examined in terms of recent advances like mobile edge computing, cloudlets, and fog computing. This has allowed academics to gain a deeper understanding of both current solutions and potential future applications. This article aims to provide a thorough overview of current developments in edge computing while emphasising the key applications. In real-world situations where response time is a crucial need for many applications, it also examines the significance of edge computing. The prerequisites and open research issues in edge computing are discussed in the article's conclusion.***

***Keywords:*** ***Smart health system, Smart meters, Smart home, Smart parking system, Smart sea monitoring, Smart transportation.***

## I. INTRODUCTION

A novel idea in the computer landscape is edge computing. It is characterised by speedy processing and quick application response time and brings the cloud computing service and utilities closer to the end user. Surveillance, virtual reality, and real-time traffic monitoring are just a few of the internet-enabled applications that are now being developed [1, 2]. These applications are typically used by end users on their resource-constrained mobile devices, with cloud servers handling the processing and essential service functions. High latency and mobility-related problems occur when mobile devices use cloud services [3, 4]. Edge computing satisfies the aforementioned application needs by moving computation to the network's edge. The three Edge computing models Cloudlets [5, 6], Fog computing [7], and Mobile Edge computing [8, 9] can be used to address the problems with cloud computing. Applications for edge computing include smart transit, smart meters, smart homes, smart parking systems, and smart sea monitoring.

The idea of mobile edge computing, where mobile users can make use of base station-based computing services, has been developed by the European Telecommunications Standards Institute (ETSI). Cisco [10] proposed the fog computing concept, which enables applications to run directly at the network edge through billions of intelligently connected devices. Cloudlets are a notion that Satyanarayanan et al.

[11] developed to address the latency issue with cloud access by utilising the local network's computing capabilities. Similar to this, mobile edge computing offers processing, application services, and storage off-loading near to the end users. Mobility support, location awareness, extremely low latency, and close closeness to the user are some of the intriguing characteristics of edge computing [12].

As shown in Figure 1, these characteristics make edge computing suited for a variety of future applications including data analytics, industrial automation, virtual reality, real-time traffic monitoring, smart homes, and smart marine monitoring. Different services, like as QoS, VPN, and Voice over IP, are hosted by edge devices like routers, access points, and base stations [13]. These Edge devices serve as a link between the cloud and mobile, intelligent devices. Several surveys have examined different facets of edge computing, such as fog computing (S. Yi et al. [14], L. M. Vaquero et al. [15], I. Stojmenovic [16], F. Bonomi et al. [17], and T.H. Luan et al. [18]), while a smaller number of studies concentrate on the edge computing domain, such as mobile edge computing, which focuses on particular application domains. There hasn't been a thorough study done yet that covers Mobile-Edge, Fog, and Cloudlet computing, among other features of Edge computing. This article's primary contribution consists of:

*1)* A thorough analysis of all aspects of edge computing (Cloudlet, Fog and Mobile-Edge).
*2)* A new classification of multi-facet computing paradigms within Edge computing.
*3)* The identification of key requirements to envision the Edge computing domain.
*4)* The exploration of open research challenges.

©IJRASET: All Rights are Reserved 

*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 11 Issue I Jan 2023- Available at www.ijraset.com*

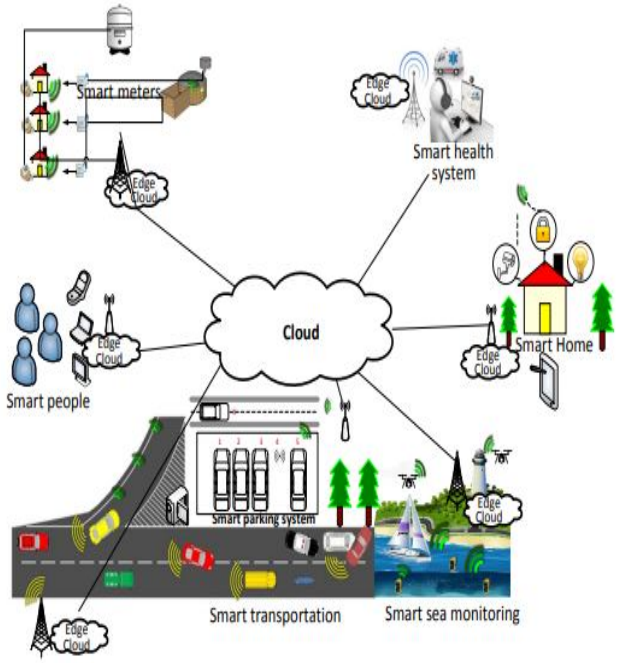
The remainder of the text is organised as follows: The fundamental ideas of cloud computing, edge computing, and related concepts, as well as their design and distinctions, are presented in Section 2. The properties of edge computing are covered in Section 3. In Section 4, the state-of-the-art in edge computing is examined, and research is compared to show the benefits and drawbacks of the current frameworks. The prerequisites for achieving the goal of edge computing are outlined in Section 5. The obstacles and open questions in research are highlighted in Section 6. The paper is finally concluded in Section 7. A list of the acronyms used in this paper is provided in Table 1.

## II. BACKGROUND

The essential ideas of cloud computing are discussed in this section, along with how it differs from edge computing. This section also covers the fundamental ideas of Cloudlet, Fog, and Mobile Edge computing. This section's goal is to give the reader a firm understanding of the study topic.

### *A. Cloud Computing*

Through a pool of computing resources that includes storage services, computing resources, and other resources, cloud computing is a paradigm for computing that provides on-demand services to end users[20]. The three main services provided by cloud computing are software as a service (SAAS), platform as a service (PAAS), and infrastructure as a service (IAAS) [21]. All of these firms offer on-demand computer capabilities, including data processing and storing [22]. Along with providing the aforementioned services, cloud computing also emphasises dynamic resource optimization for various users. For instance, a user in the west will receive an email resource from the cloud according on that user's time zone. Through cloud computing, Asian users are also given access to the same resource based on their respective time zones.




### *B. Edge Computing*

Edge computing sends computational data, applications, and services to the edge of a network rather than to Cloud servers. By providing services closer to the users, content providers and application developers can make use of edge computing systems. High bandwidth, extremely low latency, and real-time access to network data that can be exploited by numerous applications are characteristics of edge computing [23–25]. By enabling new apps and services, the service provider can make the radio access network (RAN) accessible to Edge users. For both businesses and individuals, edge computing opens up a number of new services [26]. Location services, augmented reality, video analytics, and data caching are some of the use cases for edge computing. As a result, the adoption of Edge platforms and the emerging Edge computing standards serve as crucial enablers for the creation of new revenue streams for operators, vendors, and third parties.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

### C. Main Differences Between Cloud and Edge Computing

With the use of edge computing, which is a more advanced kind of cloud computing, latency is decreased by putting services closer to the end consumers. By making resources and services available in the Edge network, edge computing lessens the burden on a cloud. However, Edge computing enhances the end-user experience for delay-sensitive applications, which makes Cloud computing more effective [27].

Similar to cloud computing, edge service providers offer end users access to application, data processing, and storage services. Despite these basic similarities, these two new computer paradigms have a number of significant distinctions. The location of the servers is where Edge computing and Cloud computing diverge most. While cloud computing services are located on the Internet, edge computing services are situated in the Edge network.

The multi-hop distance between the end user and the cloud servers determines how readily available cloud services are via the Internet.

In contrast to Edge computing, which has low latency, cloud computing has high latency due to the much greater distance between the mobile device and the cloud server. Similar to how Edge computing has extremely little jitter and Cloud computing has high jitter, In contrast to cloud computing, edge computing is highly mobile and location-aware. As opposed to cloud computing, which uses a centralised architecture, edge computing uses a dispersed model for server deployment. Due to the longer trip to the server, the likelihood of data en route attacks is higher in cloud computing than in edge computing. General Internet users are the intended users for cloud computing, whereas Edge computing's targeted service customers are Edge users. The reach of Edge computing is constrained in comparison to the worldwide reach of Cloud computing. Last but not least, Edge hardware is less scalable than the Cloud due to its limited capabilities.

### D. Edge Computing and Similar Concepts

Edge computing is a development of cloud computing in which computer services are made available at the network's edge, closer to the end users. The Edge vision was created to address the problem of high latency in delay-sensitive services and applications that aren't handled adequately within the cloud computing paradigm. Location awareness, mobility support, and extremely low and predictable latency are needed for these applications.

Even though edge computing has a number of advantages over cloud computing, research in this new field is still in its early stages. The three facets of edge computing—Cloudlets, Fog computing, and Mobile Edge computing—are all thoroughly covered in this survey essay.

[15] describes edge computing as an autonomous computing paradigm made up of a large number of distributed heterogeneous devices that interact with the network and carry out computing functions including processing and storage. These duties may also assist in providing lease-based services, in which a user rents a device and receives rewards. Fog computing, which transports resources and services from the core network to the edge network, is an extension of the cloud computing paradigm, according to Cisco [10].

It is a virtualized platform that offers networking, storage, and processing capabilities in the edge network. Some of the ideas that are similar to the fog computing paradigm are cloudlet [11] and mobile-edge computing [28]. Only mobile users are intended for cloudlet and mobile-edge computing, which provide them the freedom to use resources that are close at hand. Fog, however, depends primarily on Cisco infrastructure, which has computational capabilities in addition to its standard router and switch operations.

## III. EDGE COMPUTING CHARACTERISTICS

Many aspects of edge computing are comparable to those of cloud computing. However, the following are the distinctive qualities of edge computing that set it apart from other technologies:

### A. Dense Geographical Distribution

Edge computing deploys various computing platforms on the edge networks to bring Cloud services closer to the user [29]. The infrastructure is helped by its widespread geographic deployment in the following ways: Location-based mobility services can be facilitated by network managers without having to travel the entire WAN; big data analytics can be carried out quickly and more accurately; and the edge systems can support real-time analytics on a large scale [32, 33]. Examples include pipeline monitoring and sensor networks for environmental monitoring.

*B. Mobility Support*

Edge computing also facilitates mobility, such as the Locator ID Separation Protocol (LISP), to directly interface with mobile devices, as the number of mobile devices is continually increasing. The LISP protocol implements a distributed directory system that decouples the location identification from the host identity. The fundamental idea behind Edge computing's mobility support is the separation of the host identity from the location identity.

*C. Location Awareness*

Users of mobile devices can receive services from the Edge server that is closest to their physical location thanks to the location-awareness feature of edge computing. To locate electronic devices, users can use a variety of technologies, including wireless access points, GPS, and cell phone infrastructure. Numerous Edge computing applications, like edge-based disaster management and fog-based vehicle safety apps, can make use of this location awareness.

*D. Proximity*

In edge computing, customers have access to processing resources and services that can enhance their experience. Users can utilise network context information to make decisions about offloading workloads and using services since computational resources and services are readily available nearby. In a similar way, the service provider can use the information of mobile users to better their services and resource allocation by collecting device information and analysing user behaviour.

*E. Low Latency*

Edge computing approaches reduce latency in service access by bringing compute resources and services closer to the users. Users can run their resource-intensive and delay-sensitive apps on the resource-rich Edge devices thanks to Edge computing's low latency (e.g. router, access point, base station, or dedicated server).

*F. Context-Awareness*

Mobile devices have context awareness, which can be defined as being dependent on location awareness.

In Edge computing, the mobile device's context information can be utilised to access Edge services and make decisions about offloading [34].

In order to provide context-aware services to Edge users, context-aware network information in real time, such as network load and user location, can be employed. Additionally, the service provider can improve customer pleasure and experience quality by using context information.

*G. Heterogeneity*

The employment of various platforms, architectures, infrastructures, computation, and communication technologies by the components of Edge computing is referred to as heterogeneity (end devices, Edge servers, and networks).

The key components of end device heterogeneity include variations in software, hardware, and technology. The main causes of edge server side heterogeneity are platforms, platforms-specific regulations, and APIs. Such discrepancies cause interoperability problems, which make them a major obstacle to the successful deployment of edge computing.

The variety of communication technologies that have an impact on the delivery of Edge services is referred to as network heterogeneity.

## IV. STATE OF THE ART ON EDGE COMPUTING

The goal of this section is to critically evaluate the body of work on edge computing paradigms, such as mobile edge computing, cloudlets, and fog computing. A taxonomy of edge computing and related ideas is shown in Figure 2.

*A. Fog Computing*

Here, we categorise the literature that has been written about fog computing into four categories: big data analytics, resource management, and security and privacy.Based on their goals,Fog computing systems are contrasted in Table2.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 11 Issue I Jan 2023- Available at www.ijraset.com*

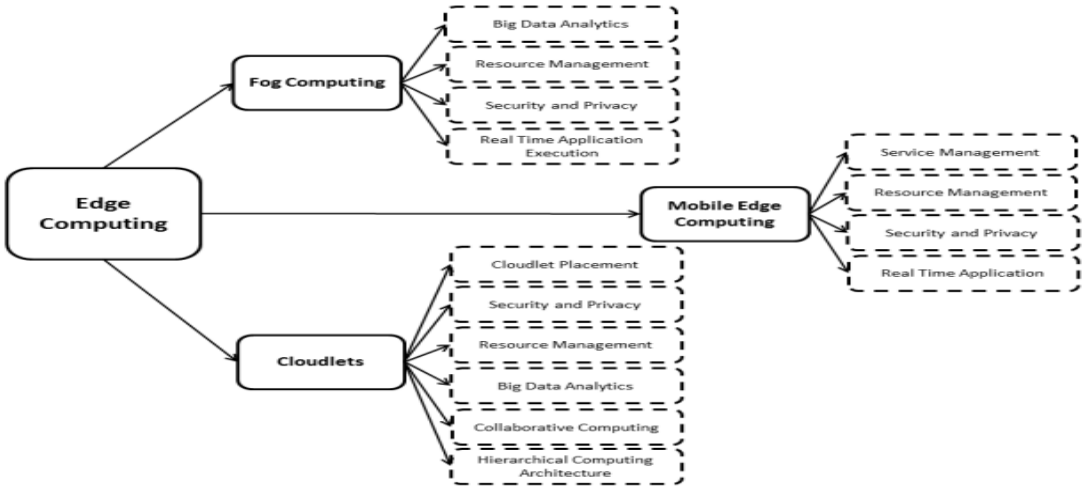



Figure 2: Classification of Edge Computing

*1) Real-time Application Execution*

An architecture for an augmented brain-computer interface (A-BCI) system was put forth by J.K. Zao et al. [35]. The online data server, also known as the Fog server, which can be placed on television set-top boxes, personal computers, or gaming consoles, is a crucial part of the design. Each Fog server functions as a data hub and signal processor. In addition, the server supports the signal's preprocessing, source identification, and fitting of an autoregressive model. Wireless sensors and the presence of the Fog server close to the user create a triangle-shaped link between the nodes. The triangle association can be used to improve the system's performance and adaptability. The suggested system also supports publish/subscribe protocols for machine-to-machine (M2M) communication. The pervasive computing system places a high priority on system security, which is implemented via the TLS protocol. Emergency Help Alert Mobile Cloud (E-HAMC), a proposed emergency alert and management service architecture by M. Aazam et al. [36], makes use of the Fog and Cloud platforms. The suggested approach effectively manages various emergency scenarios. Data is sent to the Fog server in an emergency, which then alerts the necessary emergency response departments and family members. The data is also pre-processed before being transferred to the cloud, where it is analysed and a wider range of services are created. The E-HAMC system maintains a list of immediate family members, so there is no need to look for family members' contact information in an emergency. After the user selects the event type, the application works with the Fog server to complete the remaining task. Global positioning exchanges the starting location with the Fog server. Over the Fog computing platform, M.A. Al Faruque et al. [37] delivered energy management as a service. Energy management as a service can be implemented by end users at a cheap implementation cost and short time to market thanks to the suggested platform's scalability, adaptability, and open source nature. Depending on the platform's operational domain, a variety of devices make up its hardware. Gateways, sensors, actuators, computing, and connecting devices are only a few examples of the gadgets. Multiple devices are monitored in the home energy management (HEM) platform to reduce energy use. The HEM control panel is the primary computing device in charge of managing the monitoring and controlling tasks. It is in charge of finding and monitoring the devices, managing load demands, and instructing the devices in accordance with the developed algorithms. The variety of hardware and communication is abstracted by the service-oriented architecture.

*2) Big Data Analytics*

Traditional industries have been converted into modern industries as a result of the adoption of intelligent cyber-physical systems and the spread of smart technology. In reality, the majority of factory systems rely on a centralised Cloud infrastructure, which is no longer able to handle the tremendous processing workload generated by the factory's thousands of smart devices. An technique of placing Fog computing resources close to the industrial devices was introduced in [38] to address this problem. The suggested method aids in giving sites real-time feedback. The deployment of an intelligent computing system made up of data centres, gateways, and Fog/Edge devices in a logistics centre was another key area of the study. The study also created a model for integer programming that reduces the overall installed cost of the devices with Fog support.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

Furthermore, a genetic algorithm was used to tackle the NP-hard facility location problem. High performance is confirmed by simulation results while deploying intelligent computing systems in a logistics centre. Despite the many benefits of the suggested strategy, some deployment-related ambiguous aspects were not taken into account, including data traffic, latency, energy use, load balancing, heterogeneity, fairness, and QoS. A fog-based data analytics system was proposed by R. Iqbal et al. [39] to address the difficulties in providing context-aware services in an Internet of Vehicles (IoV) environment. The data collecting layer, pre-processing layer, reconstruction layer, service layer, and application layer are the five layers that make up the framework. The first layer is in charge of gathering data from various IoV system sources, including automobiles, road sensors, autonomous vehicles, pedestrians, and roadside equipment. The pre-processing layer helps in data extraction, trimming, reconstruction, and modelling and gives the layers above a starting point for extracting valuable information from the provided datasets. The analytics layer, which is based on the Fog computing idea, is the primary layer of the system since it aids in bringing data intelligence close to the user network. While the application layer is in charge of handling various IoV apps according to their QoS requirements, the fourth layer is in charge of managing and exposing the analytics results. Although the framework's suitability was assessed using two crucial use cases, its advantages were not confirmed through implementation. A multitier Fog computing concept was suggested in [40] with the aim of providing large-scale data analytics for smart city applications. The approach includes ad hoc and dedicated fogs that alleviate the crucial latency issues related to dedicated computing infrastructure and Cloud computing. The trials were carried out on Raspberry Pi devices powered by a distributed computing engine that generates simple workload models and aids in measuring the performance of the analytics jobs. Approaches for quality-ware admission control, offloading, and resource allocation were suggested in order to optimise the benefits of the analytics service. According to the trial findings, the multitier Fog-based paradigm for smart cities considerably enhances analytics services. It was lacking, nonetheless, in a thorough assessment and validation of the model. Consumer-focused Internet of Things apps and services that support delay-sensitive operations were introduced by S. K. Datta et al. [41]. Additionally, they suggested an architecture made up of roadside units (RSUs), M2M gateways, and connected automobiles. The architecture makes it easier to provide a variety of consumer-focused services, including linked vehicle management, IoT service discovery, and data analytics using semantic web technologies. The raw data produced by sensors, smartphones, and automobiles must be processed using semantic web technology. The processed data offers useful information that connected automobiles can utilise to make wise judgements. The M2M measurement framework is installed on RSUs and M2M gateways that gather the sensor metadata and generate inferred data in order to provide this feature. The IoT services discovery module can access the services (including apps) provided by RSUs and M2M gateways. The OMA lightweight M2M technological specifications, which support high device mobility, are the foundation for the management framework for connected automobiles. The framework gives users the ability to read, write, and update sensor setups. A smart gateway (SG) architecture for fog computing was suggested by M. Aazam et al. SG is in charge of carrying out many IoT duties, including data collecting and preprocessing, data filtering, uploading only necessary data, monitoring the activities of IoT items and their energy levels, as well as data security and privacy. The base station(s) or directly to the SG can both be used to transfer the data gathered from IoT. The two types of SG-based communication are single-hop connectivity and multiple-hop connectivity. Smart items are immediately connected to the gateway that gathers data and sends it to the fog via single-hop communication. Smart sensing devices that are directly connected to the Gateway and have restricted roles should use this sort of communication. Direct contact with the SG is not feasible when several sensor and IoT networks are installed. Each sensor network has its own sink node, and the Gateway gathers data from numerous base stations and sink nodes. Large-scale IoT and mobile items are suitable for these kinds of scenarios. When computing on the edge of the user network, J. Preden et al. [43] introduced a Fog computing-based method for fusing data with the decision mechanism. This method allows for the utilisation of massive data produced by sensors while taking bandwidth limitations into account. To locate and give actionable data to decision-makers taking into account their data requirements, the data-to-decision (D2D) concept is utilised. The D2D also covers data gathering and integration of useful data to extract the appropriate situational information for assessing threats and decision-related outcomes. The authors advised using a system-of-systems method, in which each system is autonomous and has the processing power to carry out the required tasks. The D2D concept can be used to persons on the edge who have access to contemporary communication platforms, even if it is applicable to all levels of companies.

*3) Resource Management:*

Fog computing can offer a range of services in smart manufacturing since it allows for real-time analysis that can identify device flaws. Fog nodes get massive amounts of data from the manufacturing-related equipment. Fog nodes need effective resource utilisation algorithms since they have limited computational and storage resources.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

By utilising virtualization technology, resource consumption can be enhanced and the disadvantages of resource rivalry can be avoided. Despite having a lengthy startup time and expensive monitoring, virtual machines (VMs) are one of the most extensively used virtual technologies. A lightweight virtualization technology called container has been created to address these problems. The container concept in fog computing was used by the researchers in [44] to examine task scheduling issues and to propose container-based task scheduling algorithms. The outcomes showed that through reducing latency, the suggested algorithms contribute to better resource usage. The study, however, neglected the crucial aspect that cloud resources are finite, which is typically taken into account in real-world circumstances. Additionally, a significant issue that requires attention is the positioning of containers' images.

A lightweight virtualization technology called container has been created to address these problems. The container concept in fog computing was used by the researchers in [44] to examine task scheduling issues and to propose container-based task scheduling algorithms. The outcomes showed that through reducing latency, the suggested algorithms contribute to better resource usage. The study, however, neglected the crucial aspect that cloud resources are finite, which is typically taken into account in real-world circumstances. Additionally, a significant issue that requires attention is the positioning of containers' images. The study specifically concentrated on issues with effective resource management with the aim of ensuring QoS in FC-MCPS. A highly complex form of mixed-integer non-linear programming (MINLP) was used to formulate the issue. A low-complexity two-phase linear programming based heuristic approach was additionally proposed. Results from many experiments demonstrate that the algorithm works better than a greedy strategy.

To aid in the execution of mobile applications, M.A. Hassan et al. [47] presented adaptive mobile application offloading and mobile storage augmentation techniques. Active resource monitoring is necessary to make the proposed offloading decision. The choice takes into account resource dynamics and availability to forecast how certain jobs would perform when executed on various computer platforms. The suggested storage augmentation mechanism creates a distributed storage service for fog computing by combining the available storage capacity of a user's various devices. Mobile devices' resources are used for active resource monitoring, and merging their storage space necessitates a unified storage management and access system.

For new mobile apps that are geospatially dispersed and latency-sensitive, K. Hong et al. [48] presented the Mobile Fog high level PaaS development model. The orchestration of highly dynamic heterogeneous resources at different levels of hierarchy makes the creation of applications based on fog computing difficult. The complexity associated with developing Fog-based applications is reduced by the proposed PaaS programming model. To accommodate the fluctuating demands of actual applications, Mobile Fog offers on-demand allocation of compute instances. This makes it possible to employ the dynamic scaling functionality, which is dependent on a user-defined scaling policy. The scaling strategy includes conditions for scaling, such as CPU utilisation of more than 70%, as well as monitoring metrics like network bandwidth and CPU utilisation. Programming, maintenance, and debugging are all made simpler by the suggested high level PaaS programming model, which enables quicker prototyping. However, the high-level PaaS application demands more processing time and operates more slowly than the low-level programming approach.

A high-level Fog software architecture and numerous enabling technologies were provided by F. Bonomi et al. [49] and are essential for achieving the goal of fog computing. Applications from several users can coexist thanks to fog architecture, which gives each user the impression that specific resources are available to them only. The virtualization of several components, including processing, storage, and networking, is a key component of the distributed architecture of the Fog. Fog enables a mechanism that is analogous to the Cloud-based provisioning approach for autonomous resource management. Additionally, the authors discussed the crucial use cases for fog in the context of big data and the internet of things.

A Fog-based service-oriented resource management model for an IoT environment was developed by M. Aazam et al. [50]. The model tries to manage the resources effectively and fairly.

The authors divided the IoT devices into three groups while taking into account the type of the item and the mobility aspect, and they managed the resources accordingly.

The work largely concentrated on resource estimation, reservation, pricing, and forecasting while taking device and client type into account.Convex optimization problems with modelled costs and geographical diversities were the subject of C. T. Do et alstudy .'s [51]. In order to provide video streaming services in fog computing, the authors presented a plan based on shared resource allocation. Additionally, the challenge of reducing carbon footprint was examined.

Due to the significant number of Fog computing nodes involved, the joint resource allocation and carbon footprint reducing challenge is a large-scale convex optimization problem. The authors suggested a distributed solution based on a proximal method to break down the large-scale global problem into several sub-problems in order to achieve scalability and improve performance. Proximal algorithms, in contrast to conventional techniques, converge quickly and have robustness without significant assumptions.

*4) Security and Privacy*

Unstructured data has grown rapidly in recent years and is now primarily housed in cloud-based systems. Users that save their data in the cloud entirely relinquish control over it, raising the possibility of privacy leaks. Solutions for privacy protection frequently rely on encryption technology. These plans, however, are no longer sufficient to fend off attacks conducted against cloud servers. A three-layer, Fog computing-based storage system was presented in [52] as a solution to this problem. The plan takes advantage of cloud storage while guaranteeing customer data security. To calculate the distribution proportion stored in the cloud, fog, and local machine, a Hash-Solomon algorithm was also developed. The outcomes of the theoretical research confirmed that the suggested plan was workable. Fog computing uses a lot of resource allocation algorithms since they guarantee best use of the available resources. These plans, however, are susceptible to both physical and digital attacks. An approach that protects privacy was suggested in [53] as a solution to this issue. The method relies on continuous message expansion. The technique helps to mitigate against smart gateway and eavesdropping attacks, which are frequently conducted against resource allocation algorithms, according to formal analysis results. The suggested technique is also strong because it completely resists key compromise and satisfies users' privacy concerns.

Privacy-preserving Fog-enabled aggregation is the name of the smart metering aggregation framework that L. Lyu et al. [54] suggested (PPFA). The system is built on additive homographic encryption and the Fog computing architecture. By introducing concentrated Gaussian noise to their data, PPFA allows smart metres to encrypt the noisy results. Because there are implicated intermediate nodes, PPFA ensures aggregator ignorance. The use of a two-layer encryption technique ensured robustness. Aggregator obliviousness is achieved using the first layer, whereas authentication is done using the second layer. The findings showed that PPFA protects privacy by reducing network latency, reducing bandwidth bottleneck issues, and making significant energy savings. The framework must still be evaluated for distributed applications while taking into account heterogeneous queries.

A. S. Sohal et al. [55] proposed a cybersecurity framework for identifying malicious edge devices in a Fog environment by using the Markov model, intrusion detection system (IDS), and virtual honeypot device (VHD). A two-stage hidden Markov model is used for categorizing four different edge devices (legitimate device (LD), sensitive device (SD), under-attack device (UD) and hacked device (HD)) into four different levels. A secure load balancer (SLB) is used to provide services to all edge devices so that a load of Fog layer resources is balanced. The IDS is used for continuous surveillance of the services which are granted to LDs and prediction of hacked devices. The proposed cybersecurity framework works as follows: First, on the identification of any attack by IDS, attack alarm is generated, and the detection phase is triggered for that edge device. The identification and category of the attacked edge device are sent to Markov 1 which is responsible for calculating the attack probability of the edge device and its category. In the second step, Markov 2 utilizes the information of Markov 1 to predict whether the edge device should be shifted to VHD or not. If any LD is detected on VHD that is shifted mistakenly, LDP of that edge device is calculated using Markov 3, and its value is sent to Markov 4 which is further responsible for predicting whether to shift edge device back to LD status or not. VHD is responsible for storing and maintaining the log repository of all the devices identified as malicious by IDS as well as continuously monitoring the shifted edge devices to VHD. The experimental results show a better response time for achieving high accuracy. However, the important parameters of cost of response and cost of damage which affect the response time are not considered. While achieving high accuracy with a lower cost of damage, the cost of response for the proposed algorithm is high. The cost of response will increase due to the increase and severity of total number of attacks with time.

S.J. Stolfo [56] proposed an approach to secure the Cloud by utilizing decoy information technology. The authors introduced disinformation attacks against malicious insiders using the decoy information technology. Such attacks prevent the malicious insiders from differentiating the real sensitive data from customer fake data. The security of the Cloud services is enhanced by implementing the two security features of user behavior profiling and decoys. User profiling constitutes a common technique used to detect when, how, and how much a user accesses the data in the Cloud. The normal behaviour of a user can be regularly monitored to identify abnormal access to a user's data. The decoy consists of worthless data generated to mislead and confuse an intruder into believing that they have accessed the valuable data. The decoy information can be generated in several forms, such as decoy documents, honeypots, and honeyfiles. The decoy approach may be used combined with user behavior profiling technology for ensuring the information security in the Cloud. When abnormal access to the service is observed, fake information in the form of decoy is delivered to the illegitimate user in such a way that it seems completely normal and legitimate. The legitimate user immediately recognizes when decoy data is returned by the Cloud. The inaccurate detection of unauthorized access is informed by varying the Cloud responses using different means such as challenge response to the Cloud security system. The proposed security mechanism for Fog computing still requires further study to analyze false alarms generated by the system and miss detections.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

### B. Cloudlets

In this subsection the literature related to Cloudlets is classified into resource management, Big Data analytics, service management, Cloudlet placement, collaborative computing, and hierarchical computing architecture. Table 3 shows the comparative summary of Cloudlets based schemes.

#### 1) Resource Management

FemtoClouds is a dynamically self-configuring Cloudlet system that coordinates multiple mobile devices to provide computational offloading services [57]. Contrary to traditional Cloudlet phenomena, FemtoClouds utilizes the surrounding mobile devices to perform computational offloading tasks, thereby reducing the network latency. In the proposed system, a device (laptop) acting as a Cloudlet creates a Wi-Fi access point and controls the mobile devices (compute cluster) willing to share computation. Mobile devices can offload a task by giving proper information about the task (such as input and output data size and computational code) to the Cloudlet. Cloudlet first inspects the available computation resources in the compute cluster and the time required to execute the task. Then, the task is scheduled to the available devices using the greedy heuristic approached optimization model. The presented system creates a community cloud with minimal dependent on corporate equipment. The performance of the presented system depends on the number of mobile device participation in the compute cluster.

L. Liu et al. [58] proposed a two-level optimization mechanism to improve resource allocation while fulfilling the user demands in a multi-Cloudlet environment. In the first level, MILP is used to model the optimal Cloudlet selection among multiple available considering the user demand. If the available Cloudlets are unable to meet the user requirements, the request is forwarded to the remote cloud. Then, in second level, MILP is used to develop a resource allocation model for allocating resources from the selected Cloudlet. The Cloudlet selection is based on objective function of optimizing latency and mean reward whereas selection of resource allocation is based on optimizing reward and mean resource usage. Although the authors claim that the proposed system has improved latency and maximized system resources usage and reward, there is a need to study the overhead and complexity before the practical deployment of the system.

#### 2) Big Data Analytics

In order to enable the analytics at the edge of the Internet for high data rate sensors used in IoT, the GigaSight architecture was proposed in [59]. GigaSight is a repository (of crowd-sourced video content) used to enforce measures like access control and privacy preference. The proposed architecture constitutes an amalgamation of VM-based Cloudlets that also offers a scalable approach for data analytics. The aim of the study was to demonstrate that how Edge computing can help improve the performance for IoT-based high-data rate applications.

#### 3) Cloudlet Placement

L. Zhao et al. [60] proposed a network architecture for SDN-based IoT that focuses on optimal placement of Cloudlets to minimize the network access delay. The proposed architecture consists of the two components of control plane and data plane. The control plane is responsible for signaling and network management whereas data plane forwards the data packets. The SDN controller keeps the complete view of the network that helps manage the data transmission. Each access point (AP) forwards the data as guided by the SDN controller. The authors proposed enumerationbased optimal placement algorithm (EOPA) and Ranking-based Near-Optimal Placement algorithm (RNOPA) to solve the problem. In EOPA, all potential cases of k Cloudlets placement are enumerated, and the mean Cloudlet access delay of all cases are compared to obtain the optimal one with minimal average access delay. In RNOPA, those k APs collocated with Cloudlets and most fit are identified before evaluating the entropy weight of each connection feature. Although RNOPA can achieve near-optimal performance in terms of network access delay and network reliability, there is a need to investigate the stability of the Cloudlet system.

Another solution to address the problem of Cloudlet placement was proposed by M. Jia et al. [61] in the form of two heuristic algorithms for optimal Cloudlet placement in a user-dense environment of wireless metropolitan area network (WMAN). A a simple heaviest-ap first (HAF) algorithm places a Cloudlet at those APs with the heaviest workloads. However, the drawback of the HAF approach is that the APs with highest workload are not constantly nearest to their users. A density-based clustering (DBC) algorithm is proposed to overcome the shortcoming of the HAF algorithm by deploying Cloudlets in dense users' places. The algorithm also assigns mobile users to the deployed Cloudlets considering their workload. Despite the optimal Cloudlet placement, the proposed solutions do not provide support for user mobility considered as a promising feature of mobile networks.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

*4) Collaborative Computing*

F. Hao et al. [62] proposed a two-layer multi-community framework (2L-MC3 ) for MEC. The framework allocates tasks in community Clouds/Cloudlet-based environment by incorporating costs (such as monetary, access), energy consumption, security level and average trust level.

The problem is optimized in two ways: horizontal optimization and vertical optimization. The horizontal optimization maximizes the Green Security-based Trust-enabled Performance Price ratio whereas vertical optimization minimizes the communication cost between selected community Clouds and Cloudlets. Although the proposed framework minimizes the accessing cost, optimizes the degree of trust and saves the energy, the use of bi-level programming makes the solution complex.

*5) Hierarchical Computing Architecture*

Q. Fan et al. [63] proposed a hierarchical Cloudlet network architecture that allocates incoming user requests to an appropriate Cloudlet. The user's offloading request is forwarded to the tier-1 Cloudlet. If the tier-1 Cloudlet is unable to handle the request; the request is forwarded to the upper tier Cloudlet. If all tier Cloudlets are unable to handle, the request is forwarded to the cloud. A Workload ALLocation (WALL) algorithm is also proposed for the hierarchical Cloudlet network. The proposed scheme facilitates in distributing the users' workload among various levels of Cloudlets and allotting optimal resources to each user. The problem is formulated as an optimization problem that aims at minimizing the average response time by incorporating network delay and computing delay.

*6) Security and Privacy*

Z. Xu et al. [64] proposed privacy-preserving based offloading and data transmission mechanism for IoT enabled WMANs. For fast processing, offloading and transmission of IoT data, the APs with the shortest path are selected using the Dijkstra Algorithm. The IoT data transmission and task offloading are modelled as a multi-objective problem and solved using non-dominated sorting differential evolution (NSDE) technique. In order to preserve the privacy, the processing of data is offloaded to different Cloudlets based on the attributes of IoT data. The experimental results show that the proposed method has achieved better performance with lower energy consumption and transmission. However, selecting the data attributes for privacy preserving to different Cloudlets results into extra processing overhead which is not considered in evaluations of the proposed mechanism.

*C. Mobile Edge Computing*

In this subsection we have classified the literature related to Mobile Edge computing into Resource Management, Service Management, Real-Time Applications and Security Privacy. Table 4 shows the comparative summary of Mobile Edge computing based approaches.

*1) Resource Management*

S. Wu et al. [65] argued that although Mobile Cloud computing and Mobile Fog computing reduce transmission latency, the containers/virtual machines (VM) are too heavy for Edge/Fog servers due to the resource-constrained environment involving WiFi access points and cellular base stations. In order to address this problem, the authors implemented lightweight run-time offloaded codes for efficient mobile code offloading at the edge server runtime. The proposed method integrates the existing Android libraries into OSv unikernel. The OSv is a library used to convert Android application codes to unikernel. The proposed approach was evaluated using performance metrics of boot-up delay, memory footprint, image size, and energy consumption on single application with promising results. The limitations of the proposed approach include the overhead associated in converting each application to unikernel, which means that the proposed approach is not feasible for multi-process applications.

Q. Fan et al. [66] minimized the total response time of user equipment (UE) request within the network by proposing a scheme known as Application aware workload Allocation (AREA). The proposed approach works by allocating different user equipment (IoT based) requests among the least work loaded Cloudlets with minimum network delay between the UE and the Cloudlet. In order to minimize the delay, the UE requests are assigned to the nearest Cloudlets. The proposed approach was simulated on a medium size cluster using the evaluation performance metrics of response time, computing delay, and network layer with significant improvement over existing approaches. However, in real scenario, the simulation set-up seems infeasible as within an area of 25 km, setting up 25 base stations (BS) is impractical. There is high probability of BS overshooting resulting in interference within neighbouring BS, which would cause network delay.

Y. Liu et al. [67] studied the use of Edge computing for computation offloading. The aim is to achieve the better quality of service provisioning. The authors have formulated an incentive type of mechanism involving a set of interactions between the cloud service operator and the local edge server to maximize the cloud service operator utilities in the form of Stackelberg game. In Stackelberg game, the cloud server operator (CSO) acts as a leader that specify the payments to the end-server owners (ESOs) and ESO (acts a follower) that determines the computation offloaded for CSO. As this strategy makes ESOs profit-driven, this results in ESO serving only its own users, which ensures efficient computation offloading and enhanced cooperation between the ESOs. However, with the sufficiently large revenue, there will be less computation offloading by ESOs to CSO due to low utility.

W. Chen et al. [68] addressed the problem of computation offloading in Green Mobile Edge Cloud Computing (MECC). Since MECC is composed of wireless devices (WDs), the work load from a mobile device (MD) can be computed using WDs. A multi-user multi-task computation offloading framework for MECC was proposed based on a scheduling mechanism to map the workload from MD to multiple WDs. The Lyaponuv optimization approach was used for determining the amount of energy to be harvested at WDs and incoming computation offloading requests into MECC along with WDs assigned to compute the workload for the admitted incoming offloading request. This process maximizes the overall system utility.

R. Wang et al. [69] proposed a knowledge-centric cellular network (KCE) architecture to maximize the network resource utilization in learning-based device-to-device (D2D) communication systems such as social and smart transportation D2D networking systems. The KCE architecture is composed of three layers, namely the physical layer, the knowledge layer, and the virtual management layer. The physical layer collects and manages the user data among the users, the knowledge layer manages the connections between users in D2D communication system, and the virtual management layer is responsible for resource allocation in the D2D assisted network. The proposed approach seems promising in vehicle-to-vehicle communication application (V2V) whereby information can be exchanged through Mobile Edge nodes.

X. Chen et al. [70] proposed a system model for cooperative Mobile Edge computing that combines local device computation and networked resource sharing. Trustworthy cooperation is achieved between different users (e.g. mobile and wearable devices users) through their social relationships developed using the device social graph model. Cooperation among devices helps in processing and execution of different offloaded tasks and network resource sharing using the bipartite matching based algorithm. The processing power of local mobile devices and available network resources in close proximity are utilized for a different type of task executions schemes including, local and offloaded tasks such as D2D, D2D assisted cloud and direct cloud. The experimental results show cost reduction in computation and tasks offloading. However, in simulation, the maximum number of devices is set to be 500 and for each trial the simulations are run only 100 times, which is insufficient for obtaining a reliable result. For the successful practical implementation of the proposed algorithm, the protocol must run for average 10,000 times and involve large social communities of devices ($\geq 10,000$).

J. Xing et al. [71] proposed the distributed multi-level storage (DMLS) model and multiple factors least frequently used (mFlu) replacement algorithm to address the emerging problem of limited storage and data loss within Edge computing. The proposed model uses intelligent terminal devices (ITD's) such as sensors, actuators as storage devices within the edge up to nth level from user-end to cloud. Once, the storage capacity of the single edge node is exhausted, the mFLU replacement algorithm takes the part of some data from that node and transfers them to the upper nodes. The mFLU algorithm is based on traditional replacement algorithms. For evaluation purposes, a simulation was carried out by creating six level edge storage system (having levels E, S1, S2, S3, S4 and cloud) using a bottom-top approach. The proposed approach seems promising for small scale clusters (like 31 nodes in this study) due to stable upper nodes and fixed data block size. However, the data loss may be higher within large clusters due to the frequent instability of the upper nodes and the variable data block size.

G. Jia et al. [72] addressed the core issue of data storage and retrieval within the Edge computing paradigm. A new cache policy namely Hybrid-Least, a recently used cache (LRU) based on PDRAM memory architecture to solve the issues caused by phase change memory (PRAM-a non-volatile, high-density storage), was proposed. PRAM exhibits the severe drawback of limited service life compared to DRAM although it helps safeguard the data. Hence, a hybrid approach based on PRAM and DRAM was proposed. The approach works by identifying the cache block area and places DRAM blocks at the back of the cache list and PRAM blocks at the head of the cache list. It extends the existing LRU cache policy where the most frequently used data is kept at the top and least used data is pushed to the tail until it is replaced. The results show improvement of 4.6 % on PDRAM architecture. However, there is still room to improve the utilization of PRAM that achieved only 11.8 % in this study.

J. Yang et al. [73] proposed an Edge computing-based data exchange accounting system for smart IoT toys. The prototype was developed using Hyper Ledger Fabric-based block chain system.

The data exchange and the payments are done between data demanders and suppliers through consortium block chain based smart contracts. The consensus and validations of data exchange records and billing among peers are also performed through smart contracts. The proposed accounting system is responsible for safe and reliable interactions among smart toys and other IoT devices. The architecture of Edge computing-based data exchange (EDEC) consists of the five steps of member registration, data products release, order generation, data transmission, and accounting and payment. All transactions between the suppliers and demanders are recorded in four different logs in a distributed fashion and stored on EDCE servers in a centralized manner. For better performance and to support a high number of requests (high throughput), the proposed prototype system requires more rigorous experimental evaluation.

*2) Service Management*

S. Wang et al. [74] discussed the quality of service (QoS) issue for context-aware services such as service recommendation. The authors highlighted the importance of QoS prediction in the service recommendation system during user mobility. They proposed a service recommendation approach based on collaborative filtering to make QoS prediction possible using user mobility. The proposed scheme works by calculating the user/edge server similarity based on the user's changing locations. Finally, to decrease data volatility, Top-K most similar neighbors are selected for QoS prediction. The proposed approach shows improved results compared to previous approaches. It is suitable when the density of the Edge server nodes is sparse, yet for larger networks involving more Edge servers, the similarity computation might produce incorrect results.

*3) Real-Time Applications*

Z. Zhao et al. [75] proposed the three-phase deployment approach for reducing the number of Edge servers to improve throughput between IoT and Edge nodes. The aim is to support realtime processing for a large-scale IoT network. The proposed approach comprises of discretization, a utility metric, and the deployment algorithm. In the discretization phase, the whole network is discretized into small sections, and the candidate node is identified from the centroid of each section. The performance gained of each candidate node along with its link quality and correlation is evaluated in utility metric phase. Finally, the best node is deployed within the network resulting in maximum throughput. The approach was evaluated using the performance metric of throughput with significant improvement in throughput for heterogeneous data. However, in case of applications requiring homogenous amount of data, the proposed approach may not be suitable due to the accountability of traffic diversity parameter.

M. Chen et al. [76] proposed an Edge and cognitive computing (ECC) based smart healthcare system for patients in emergency situations. Cognitive computing is utilized for analyzing and monitoring the physical health conditions of patients in emergency situations. Based on the health condition of the patient, the processing resources of Edge devices are allocated. Cognitive computing processing and analysis are performed on Edge devices for low latency and fast processing. The proposed ECC system architecture consists of the two modules of data and resource cognitive engines. The role of data cognitive engine is to collect and analyze the data. Different types of internal and external data are collected by the data cognitive engine, which includes patient physical and behavioural data, type of network, data flow, and communication channel quality. The resource cognitive engine is responsible for collecting the information related to the available Edge, Cloud, and network resources before being sent to the data cognitive engine for the allocation of resources according to the requirement and risk level of the patient. However, systems related to healthcare face the challenge of user privacy. Furthermore, the proposed system lacks privacy protections of sensitive user data. The proposed system calculates the health risk level of each user based on his or her sensitive information, which is stored and processed on Edge systems that may leak.

## V. EDGE COMPUTING OPEN CHALLENGES

The complexity of these Edge computing systems has given rise to a number of technical challenges, such as mobility management, security and privacy, scalability, heterogeneity, reliability, and resource management. There are some of the other important challenges that are still needed to be resolved. Table 5 contains the open challenges and factors along with potential guidelines to resolve the mentioned challenge.

### *A. User's Trust on Edge Computing Systems*

The success of any technology is positively linked with consumer acceptance. Trust is regarded as one of the most important factors for the acceptance and adoption of these Edge systems by the users.

As highlighted in the literature, security and privacy are among the basic challenges faced by Edge computing systems/ technologies. Since trust of a consumer is closely associated with the security and privacy of the technologies and thus, if security and privacy of user0 s data is not addressed well, it will definitely shatter the trust of a consumer, leading to non-acceptance of these edge systems/ technologies. Therefore, research efforts are required to develop consumer trust models for the adoption of Edge computing systems. A recently proposed consumer trust model [87, 88] highlights the influential factors (including security and privacy requirements) that stimulate the consumer's trust for adoption and usage of IoT products that can be applied in Edge computing systems.

*B. Dynamic and Agile Pricing Models*

The rapid growth of the Edge computing paradigm has opened up the need for dynamic pricing models that can meet the changing expectations by striking a right balance between the customer's expectations of quality of service, less delay and price, and the service provider's cost and operational efficiency.

It is a challenging task to develop dynamic and agile pricing models as one pricing model may not be successful for the engagement of multiple customers.

It is also challenging to provide best-fit pricing models for heterogeneous Edge computing systems that can offer mutual benefits for service providers and customers. However, the pricing model for cloud services [89] such as "pay-as-you-go" can be used for developing dynamic pricing models for Edge computing systems.

*C. Service Discovery, Service Delivery and Mobility*

Edge computing-as-service helps the individual Edge computing system operators and service providers to fulfil their customer demands with limited processing and memory resources, lower revenue and infrastructure cost. The service discovery in distributed Edge computing systems constitutes a challenging task given the increasing number of mobile devices that require services simultaneously and uninterruptedly. This task becomes more challenging when delay is involved in discovering and selecting the other available services and resources. The automatic and usertransparent discovery of appropriate Edge computing nodes according to required resources in heterogeneous Edge computing systems also poses a challenging task for service discovery mechanisms. However, service discovery solutions proposed for peer-to-peer networks [90, 91] can help in the design and development of effective user-transparent solutions for Edge computing systems.

Seamless service delivery constitutes an important mechanism that ensures the uninterrupted and smooth migration of running an application between different Edge computing systems while the consumer is moving [92]. Seamless service delivery with mobility also poses a challenging task as mobility severely affects the different network parameters (latency, bandwidth, delay, and jitter) which ultimately results in application performance degradation. In addition, the non-accessibility of local resources for a mobile user from outside the network due to the implementation of different security policies and billing methods also makes seamless service delivery challenging. The seamless service delivery in the context of mobility poses a vital research problem that needs to be addressed.

Effective mobility management mechanisms capable of discovering available resources in a seamless manner are required for supporting the seamless service delivery with mobility. In order to overcome this issue, seamless handoff and mobility management solutions [93–95] for wireless networks, can be used for effective design and development of seamless service delivery mechanisms in Edge computing systems.

*D. Collaborations between Heterogeneous*

Edge Computing Systems The ecosystem of Edge computing systems consists of a collection of different heterogeneous technologies such as Edge data servers (data centres) and different cellular networks (3G, 4G and 5G).

Although this heterogeneous nature of the Edge computing network allows Edge devices to access services through multiple wireless technologies such as WiFi, 3G, 4G and 5G, it makes the collaboration between such multi-vendor systems a challenging task [96].

Also, interoperability, synchronization, data privacy, load balancing, heterogeneous resource sharing and seamless service delivery are among the factors which make collaboration between heterogeneous Edge computing systems challenging.

The research efforts for interoperability and collaborations in ubiquitous systems, such as reported in [97] can be used for designing and developing efficient collaboration techniques among heterogeneous Edge computing systems.

### *E. Low-Cost Fault Tolerant Deployment Models*

Fault tolerance ensures the continuous operation of any system in the event of failure with little or no human involvement. In Edge computing systems, fault tolerance is achieved through fail-over and redundancy techniques in order to guarantee the high availability of services, data integrity of critical business applications, and disaster recovery of the system in case of catastrophic events by using servers in different physical locations, backup power supply batteries (UPS) and equipment with harsh environmental resistance. However, it is very challenging to provide low-cost fault tolerance deployment models in Edge computing since a remote backup server requires high bandwidth and additional hardware which is very costly. Machine learning-based anomaly detection or predictive maintenance-based systems for power supply batteries/UPS system constitute a cost effective solution. Predictive maintenance will avoid unscheduled downtime and will effectively reduce the cost required for backup/redundant batteries and UPS.

### *F. Security*

Although Edge computing has revolutionized Cloud computing systems by tackling latency issues, it has brought along imperative challenges, especially in terms of security. Ensuring Edge computing security has become a major challenge due to its distributed data processing. Moreover, the distinctive characteristics of Edge computing such as location-awareness, distributed architecture, requirement of mobility support and immense data processing, hinder the traditional security mechanisms to be adopted in the Edge computing paradigm [98]. Security threats associated with Edge computing can be classified into personal point of view (e.g., the user, network operator, and third-party application provider); attribute point of view, (e.g., privacy, integrity, trust, attestation, verification, and measurement); and compliance, dealing with lawful access to data and local regulations1 . In order to cope with the aforementioned security issues in Edge computing, numerous indispensable solutions with respect to identity and authentication, access control systems, intrusion detection systems, privacy, trust management, visualization, and forensics, have to be developed. The intrinsic characteristics of block chain technology such as tamper-proof, redundant, and self-healing [99–101], can help meet certain security objectives while it engenders new challenges that have to be addressed. Additionally, quantum cryptography based solutions can also be adopted in the Edge computing paradigm.

## VI. CONCLUSION

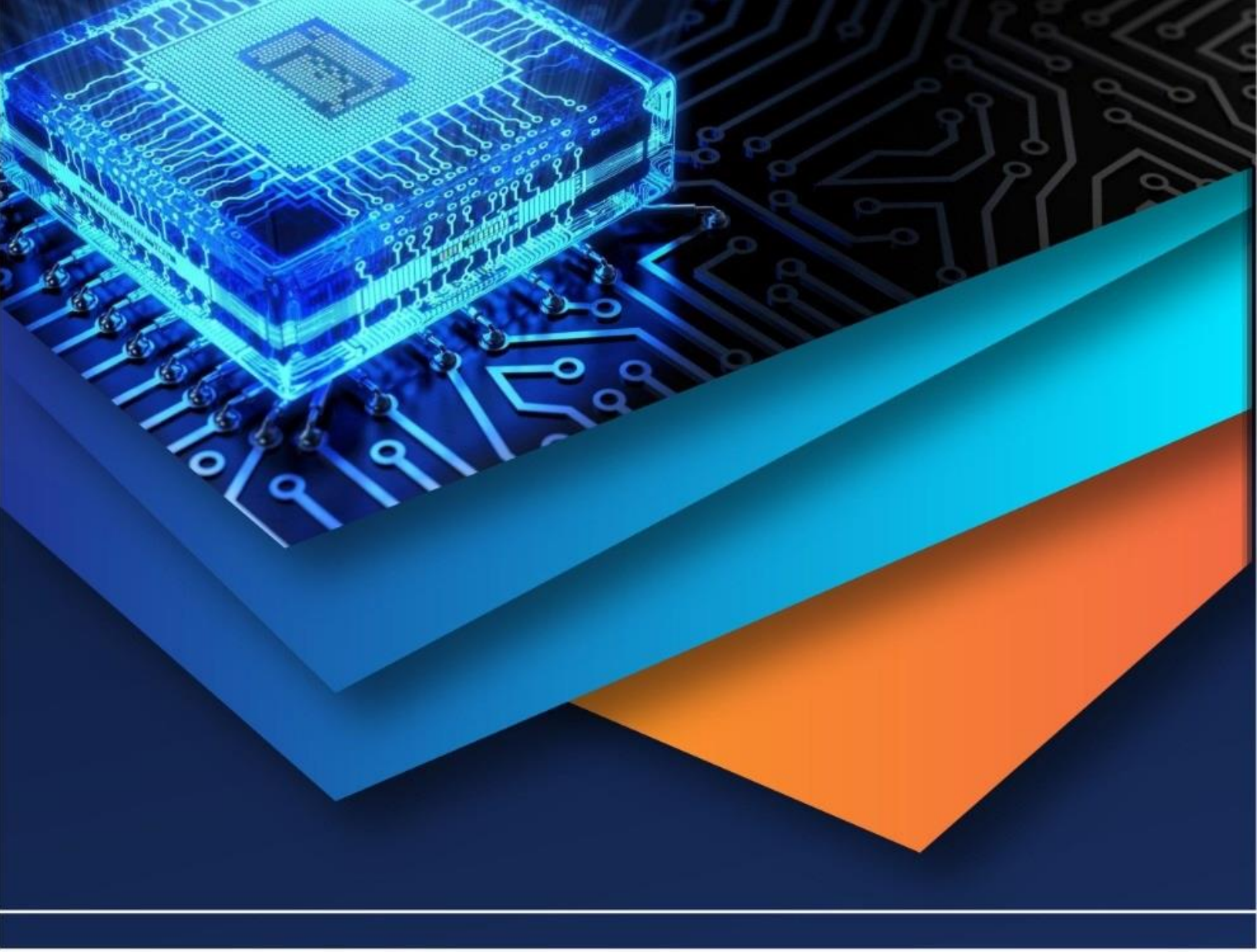
Edge computing envisions to bring services and utilities of Cloud computing closer to the end user for ensuring fast processing of data-intensive applications. In this paper, we comprehensively studied the fundamental concepts related to Cloud and Edge computing. We categorized and classified the state-of-the-art in Edge computing (Cloudlets, Fog and Mobile Edge computing) according to the application domain. The application domain areas include services such as real-time applications, security, resource management, and data analytics. We presented key requirements that need to be met in order to enable Edge computing. Furthermore, we identified and discussed several open research challenges. This study concludes that the state-of-the-art in Edge computing paradigm suffers from several limitations due to imperative challenges remaining to be addressed. Those limitations can be compensated by proposing suitable solutions and fulfilling the requirements such as dynamic billing mechanism, real time application support, joint business model for management and deployment, resource management, scalable architecture, redundancy and fail-over capabilities, and security. This study serves as an excellent material to future researchers to comprehend the Edge computing paradigm and take the research forward to resolve the unaddressed issues. Our future research aims to explore the research trends in Multi-access Edge computing networks.

## REFERENCES

[1] N. Hassan, S. Gillani, E. Ahmed, I. Yaqoob, M. Imran, The role of edge computing in internet of things, IEEE Communications Magazine (99) (2018) 1–6.
[2] M. Liu, F. R. Yu, Y. Teng, V. C. Leung, M. Song, Distributed resource allocation in blockchain-based video streaming systems with mobile edge computing, IEEE Transactions on Wireless Communications 18 (1) (2019) 695–708.
[3] E. Ahmed, A. Akhunzada, M. Whaiduzzaman, A. Gani, S. H. Ab Hamid, R. Buyya, Networkcentric performance analysis of runtime application migration in mobile cloud computing, Simulation Modelling Practice and Theory 50 (2015) 42–56.
[4] P. Pace, G. Aloi, R. Gravina, G. Caliciuri, G. Fortino, A. Liotta, An edge-based architecture to support efficient applications for healthcare industry 4.0, IEEE Transactions on Industrial Informatics 15 (1) (2019) 481–489.
[5] U. Shaukat, E. Ahmed, Z. Anwar, F. Xia, Cloudlet deployment in local wireless networks: Motivation, architectures, applications, and open challenges, Journal of Network and Computer Applications 62 (2016) 18–40.
[6] I. Yaqoob, E. Ahmed, A. Gani, S. Mokhtar, M. Imran, S. Guizani, Mobile ad hoc cloud: A survey, Wireless Communications and Mobile Computing 16 (16) (2016) 2572–2589.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

[7] W. Bao, D. Yuan, Z. Yang, S. Wang, W. Li, B. B. Zhou, A. Y. Zomaya, Follow me fog: Toward seamless handover timing schemes in a fog computing environment, IEEE Communications Magazine 55 (11) (2017) 72–78. doi:10.1109/MCOM.2017.1700363.
[8] E. Ahmed, M. H. Rehmani, Mobile edge computing: opportunities, solutions, and challenges, Future Generation Computer Systems 70 (2017) 59–63.
[9] Y. Jararweh, A. Doulat, O. AlQudah, E. Ahmed, M. Al-Ayyoub, E. Benkhelifa, The future of mobile cloud computing: integrating cloudlets and mobile edge computing, in: Telecommunications (ICT), 2016 23rd International Conference on, IEEE, 2016, pp. 1–5.
[10] Cisco fog computing solutions: Unleash the power of the internet of things,available at :https://www.cisco.com/c/dam/en_us/solutions/ trends/ iot/docs/ computing-solutions.pdf, 2015 (Accessed on 23 July 2018).
[11] M. Satyanarayanan, P. Bahl, R. Caceres, N. Davies, The case for VM-based cloudlets in mobile computing, IEEE Pervasive Computing 8 (4) (2009) 14–23. doi:10.1109/mprv.2009.82.
[12] A. Ahmed, E. Ahmed, A survey on mobile edge computing, in: 10th International Conference on Intelligent Systems and Control (ISCO), 2016, pp. 1–8. doi:10.1109/ISCO.2016.7727082.
[13] Z. Zhang, W. Zhang, F. Tseng, Satellite mobile edge computing: Improving qos of high-speed satellite-terrestrial networks using edge computing techniques, IEEE Network 33 (1) (2019) 70–76. doi:10.1109/MNET.2018.1800172.
[14] S. Yi, C. Li, Q. Li, A survey of fog computing: concepts, applications and issues, in: Proceedings of the 2015 workshop on mobile big data, ACM, 2015, pp. 37–42.
[15] L. M. Vaquero, L. Rodero-Merino, Finding your way in the fog: Towards a comprehensive definition of fog computing, ACM SIGCOMM Computer Communication Review 44 (5) (2014) 27–32.
[16] I. Stojmenovic, S. Wen, The fog computing paradigm: Scenarios and security issues, in: Computer Science and Information Systems (FedCSIS), 2014 Federated Conference on, IEEE, 2014, pp. 1–8.
[17] F. Bonomi, R. Milito, J. Zhu, S. Addepalli, Fog computing and its role in the internet of things, in: Proceedings of the first edition of the MCC workshop on Mobile cloud computing, ACM, 2012, pp. 13–16.
[18] T. H. Luan, L. Gao, Z. Li, Y. Xiang, G. Wei, L. Sun, Fog computing: Focusing on mobile users at the edgearXiv:1502.01815v3.
[19] E. Ahmed, A. Ahmed, I. Yaqoob, J. Shuja, A. Gani, M. Imran, M. Shoaib, Bringing computation closer toward the user network: Is edge computing the solution?, IEEE Communications Magazine 55 (11) (2017) 138–144.
[20] P. Mell, T. Grance, The nist definition of cloud computing, available at: http://csrc.nist. gov/publications/nistpubs/800-145/SP800-145.pdf, 2011 (Accessed on 23 July 2018).
[21] S. Hakak, S. A. Latif, G. Amin, A review on mobile cloud computing and issues in it, International Journal of Computer Applications 75 (11).
[22] I. Yaqoob, E. Ahmed, A. Gani, S. Mokhtar, M. Imran, Heterogeneity-aware task allocation in mobile ad hoc cloud, IEEE Access 5 (2017) 1779–1795.
[23] P. Wang, C. Yao, Z. Zheng, G. Sun, L. Song, Joint task assignment, transmission and computing resource allocation in multi-layer mobile edge computing systems, in press, IEEE Internet of Things Journal, 2019.
[24] Y. Sahni, J. Cao, L. Yang, Data-aware task allocation for achieving low latency in collaborative edge computing, in press, IEEE Internet of Things Journal, 2019.
[25] J. Ren, Y. He, G. Huang, G. Yu, Y. Cai, Z. Zhang, An edge-computing based architecture for mobile augmented reality, in press, IEEE Network (2019) 12–19.
[26] Z. Ning, X. Kong, F. Xia, W. Hou, X. Wang, Green and sustainable cloud of things: Enabling collaborative edge computing, IEEE Communications Magazine 57 (1) (2019) 72–78.
[27] W. Li, Z. Chen, X. Gao, W. Liu, J. Wang, Multi-model framework for indoor localization under mobile edge computing environment, in press, IEEE Internet of Things Journal, 2019.
[28] M. Patel, et al., Mobile edge computing, available at: https://portal.etsi.org/Portals/ 0/TBpages/MEC/Docs/Mobile-edge_Computing_-_Introductory_Technical_White_ Paper_V1%2018-09-14.pdf, 2010 (Accessed on 23 July 2018).
[29] M. Satyanarayanan, How we created edge computing, Nature Electronics 2 (1) (2019) 42.
[30] E. Ahmed, I. Yaqoob, I. A. T. Hashem, I. Khan, A. I. A. Ahmed, M. Imran, A. V. Vasilakos, The role of big data analytics in internet of things, Computer Networks 129 (2017) 459–471.
[31] H. Khelifi, S. Luo, B. Nour, A. Sellami, H. Moungla, S. H. Ahmed, M. Guizani, Bringing deep learning at the edge of information-centric internet of things, IEEE Communications Letters 23 (1) (2019) 52–55.
[32] S. Jo˚A¡ilo, G. DA¡n, Selfish decentralized computation offloading for mobile cloud computing ˜ in dense wireless networks, IEEE Transactions on Mobile Computing 18 (1) (2019) 207–220.
[33] A. Ferdowsi, U. Challita, W. Saad, Deep learning for reliable mobile edge analytics in intelligent transportation systems: An overview, IEEE Vehicular Technology Magazine 14 (1) (2019) 62–70.
[34] B. Han, S. Wong, C. Mannweiler, M. R. Crippa, H. D. Schotten, Context-awareness enhances 5g multi-access edge computing reliability, in press, IEEE Access, 2019.
[35] J. K. Zao, T. T. Gan, C. K. You, S. J. R. M´endez, C. E. Chung, Y. Te Wang, T. Mullen, T. P. Jung, Augmented brain computer interaction based on fog computing and linked data, in: Intelligent Environments (IE), 2014 International Conference on, IEEE, 2014, pp. 374–377.
[36] M. Aazam, E.-N. Huh, E-hamc: Leveraging fog computing for emergency alert service, in: Pervasive Computing and Communication Workshops (PerCom Workshops), 2015 IEEE International Conference on, IEEE, 2015, pp. 518–523.
[37] M. A. Al Faruque, K. Vatanparvar, Energy management-as-a-service over fog computing platform, IEEE Internet of Things Journal 3 (2) (2016) 161–169.
[38] C. Lin, J. Yang, Cost-efficient deployment of fog computing systems at logistics centers in industry 4.0, IEEE Transactions on Industrial Informatics 14 (10) (2018) 4603–4611.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)