Open Source Network Optimization Tools for Edge Intelligence

[1]Gregory Wang and [2]David Steeg
[1,2] School of Engineering, University of Southern California, USA.
[1]wangviterbi@hotmail.com

Article Info
Journal of Computing and Natural Science (http://anapub.co.ke/journals/jcns/jcns.html)
Doi: https://doi.org/10.53759/181X/JCNS202202009
Received 08 December 2021; Revised form 20 March 2022; Accepted 25 March 2022.
Available online 05 April 2022.

Abstract – It is indeed possible to bring analysis and information storage closer to where the information is generated by implementing an edge computing model. Response times should improve while bandwidth use is reduced as a result." A common misconception is that "edge" and "IoT" are synonymous. Using edge computing in the Internet of Things (IoT) is an example of this type of distributed computing, which is sensitive to configuration and location." Instead, then alluding to a specific piece of technology, the word refers to an overall architecture. In order to discover novel study opportunities and aid users in selecting more suitable edge computing advancements, this paper provides an analysis of existing open-source computing projects. Also, a comparison of the project's applicability will be defined.

Keywords – Edge Computing, Internet of Things, Machine Learning, Artificial Intelligence.

I. INTRODUCTION

Edge computing initiatives in the past were often built with integrated software and hardware specialised function infrastructure devices. This led to vendor lock-in, restricted scalability, and increased lead times for new services to reach the marketplace. More and more network operators are turning to COTS (Commercial Off-The-Shelf) hardware-based solutions for edge technologies as a result of the advancements in cloudification and virtualization solutions. The bulk of unaffiliated application developers currently have commercial solutions based on these concepts. In addition, a burgeoning open-source culture, networks, and independent developers are fostering industry development and creating new cloud and edge technology solutions. OpenNESS (Intel Smart Edge Open), which is implemented on the system, is also widely endorsed by Intel's roadmap. This guarantees that all the newest productivity and functionality Intel technology capabilities are included in the bundle and renders the system prospective as well.

Enterprise open source and edge computing are intertwined and interdependent phenomena. To offer web and adaptive streaming from network edge located near the user, content distribution networks were established in the late 1990s. Networks in the early 2000s expanded to house hardware and software at the edge devices, leading to the first commercialized edge computing systems that featured products such as dealership locators and shopping carts. Edge relies on open hybrid cloud, which provides business IT with flexibility and consistency benefits. Rosa Guntrip, executive principal marketing director, cloud systems at Red Hat, has remarked that the connection is symbiotic [1].

To be able to respond swiftly and with the least amount of interruption, businesses need "flexibility in where they deploy their workloads" and "uniformity of operations" - for ITOps as well as developers. According to its definitions, edge computing encompasses a range of infrastructures, including network edge, data centre, and cloud (whether on-premises, in a public cloud, or both.) To put it another way: An edge approach is the awareness that pushing all computing systems into the core is not always the best approach. An approach that focuses primarily on centralization is generally ineffective for a variety of reasons, including bandwidth, delay, and resilience. However, it is crucial to maintain a high level of consistency. Open-source systems and integration tools play a role in helping to maintain this level of uniformity. Instead of deploying technology in segregated islands, the objective should be to avoid creating edge installations.

Approximately 34% of respondents said that they only utilise one public cloud, indicating that a blended or multi-cloud strategy is the norm. The Cloud Native Computational Foundation's CNCF Survey 2020 indicated that the number of firms utilising on-premises infrastructures increased faster than the number of enterprises using cloud services between 2019 and 2020 [2]. Despite the difficulty in interpreting some of these data, the broad patterns are worth mentioning. Data is a major component of the digital revolution. ' Another key area of workplace open source developing technology is data analysis with Artificial Intelligence/Machine Learning (AI/ML). Artificial Intelligence and Machine Learning (AI/ML) is driving the development of architectures that do not need significant amounts of data to be sent back to a central place.

The distribution of capabilities closer to programs and customers may also help enhance responsiveness and functionalities. Health care is an area where factors like these may have a significant impact. Additionally, this link between edge technology and digitalization draws in open-source software from enterprises. In the past, I've addressed the direct link between edge and corporate open-source software. Digitalization is also linked to business open source. There has been an 11-point increase in the percentage of respondents saying that digitalization is a significant usage of open source in the previous two years (54 percent). There has been a general growth in the importance of open source in the workplace. Rather than just being a way to save money on infrastructures, IT executives are increasingly seeing open source in the workplace as a tool that can be used for strategic projects like digital transformation.

There's an overall congruence between the qualities of business open source and the requirements of edge computing [3]. ' In other words, 84% of respondents believe that open source is a critical component of their organization's security strategy Open-source software is seen as either "safer" or "as secure" by 87 percent of respondents. Edge computing and the Internet of Things (IoT) are hotbeds of security, and this is true across the board in corporate IT. Other high-ranking characteristics are likewise obvious needs for an advantage. Eighty-one percent of those polled responded that enterprise open source gives them the freedom to tailor their solutions to their own requirements. More than 80 percent of respondents said that open-source software is critical to an organization's capacity to take advantage of cloud infrastructures and eases the adoption of a hybrid cloud strategy (78 percent). For a hybrid cloud to exist, these are essential prerequisites for creating an edge infrastructure (see **Fig. 1)**.

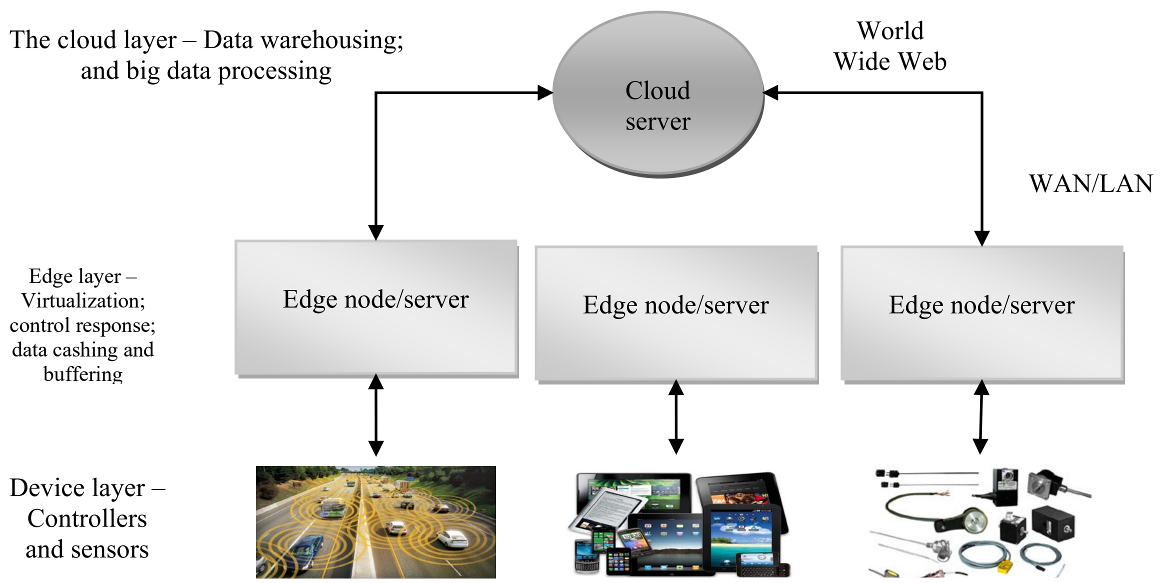

Fig 1. Infrastructure of edge computing

IT decision makers are increasingly looking for access to innovations, which is a key advantage of business open source. Corporate open source was rated by 82% of those polled as being utilised by the most forward-thinking businesses. Among the most promising areas of technological development is that of "edge computing." Edge technology is expected to generate $250.6 billion in revenue by 2024, growing at a CAGR of 12.5% between 2019 and 2024, according to market research firm International Data Corporation (IDC) [4]. Businesses may use corporate open source to assist them succeed in their edge strategy.

Edge computing technologies intended for a particular purpose, as well as certain open-source initiatives, have lately emerged. Both Akraino Edge Statck and EdgeX Foundry were released in 2017 by the Linux Foundation. A new initiative called Central Office Re-architected as a Data (CORD) centre was initiated by the Open Network Foundation (ONF). Apache Edgent was released by the Apache Software Foundation (ASF). Azure IoT Edge (AIE) was released by Microsoft in 2017 and made open source in 2018. Akraino Edge Stack (AES) and CORD offer edge cloud solutions, whereas Apache Edgent and EdgeX Foundry concentrate on IoT and try to overcome challenges that prevent practical implementations of edge computations in IoTs, Azure IoTs Edge presents hybrid edge analytics that aids in the migration of cloud technologies to IoT devices.

This paper provides an analysis of the Open-Source Edge Computing (OSEC) projects; and comparison with aspects such as primary function of the technologies, application domain, deployments, target user, system feature, virtualization, restrictions, portability, and scalability. Section II provides a critical analysis of CORD, AIE, AES, Apache Edgent, and EdgeX Foundry. Section III presents a comparison of the projects with the said aspects. Lastly, Section IV draws conclusions to the whole research.

II. CRITICAL ANALYSIS

This section provides a critical analysis of AIE, CORD, AES, Apache Edgent, and EdgeX Foundry.

Azure IoT Edge (AIE)

AIE [5] is a cloud analytics solution that aims to provide insights to edge sensors. Depending on their capabilities, these devices might be anything from routers to gateways. Using the same program architecture as other Azure IoT cloud services, users may migrate their current apps from the cloud to the edge sensors for faster response time. The ease of use makes it easier to design edge apps. Complex activities like AI technology, computer vision, and image analysis may be deployed on smart objects using Azure technologies such as Azure Functions, Azure Stream Analytics, and Azure Machine Learning,

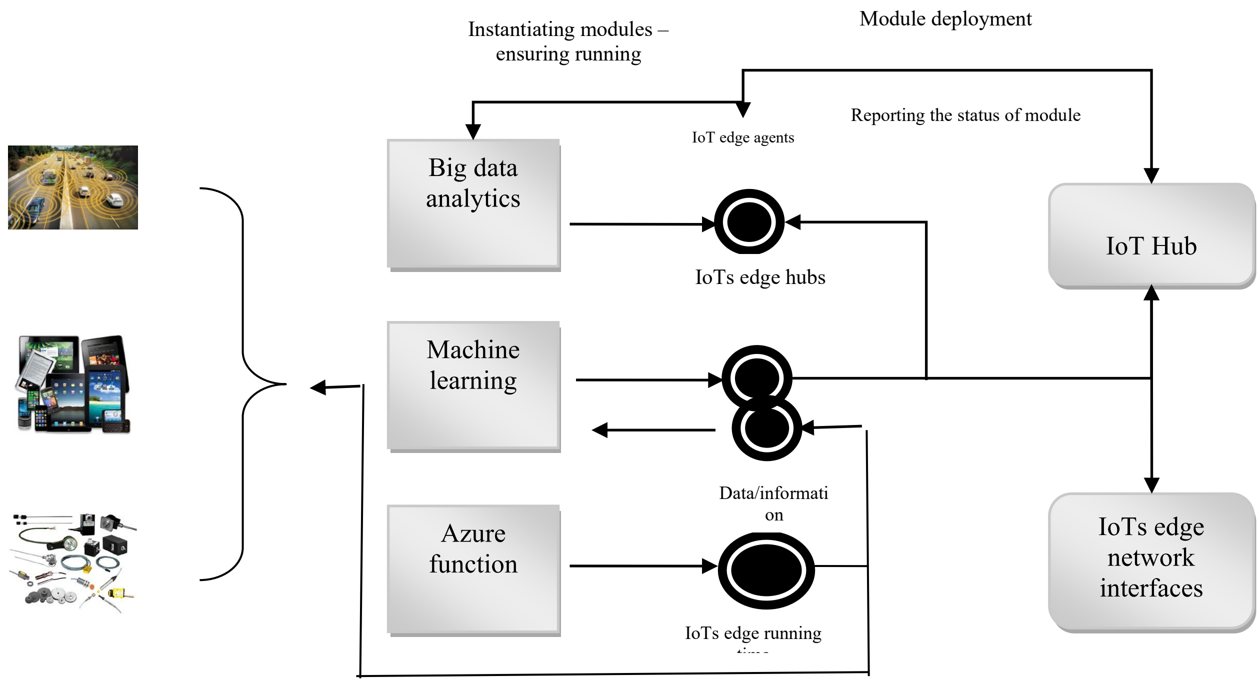

Fig 2. Representation of Azure IoTs edge

Azure Edge modules, running time, and a cloud-based interfaces are all part of the IoT Edge platform represented in **Fig. 2.** A gateway in the cloud connects the first two major components, which operate on edge devices. Customers' code or Azure services operate in containerized instances in IoT Edge components. IoT Edge runtime is in charge of the management of these subsystems. Both the edge sensors and the cloud-based interfaces are monitored and managed using the cloud-based interfaces. In simple terms, IoT Edge module acts as operation units for a particular application. Docker containers, a module instance runs the module image as a single computational unit. The same Azure applications or bespoke applications may be executed on edge devices using the same standard programming language as in the cloud if adequate resources are available. Additionally, these modules are Azure-deployable The IoT Edge may be scaled.

Edge devices are managed by the IoT Edge running time. Both IoT edge agent and IoT edge gateway are included. In the cloud system, IoT Hub is a centralized repository that serves as a central messaging node. IoT edge hubs serve as a message broker, facilitating communication between modules and the transfer of information to IoT Hub. Using the IoT Edge agents, components may be installed and monitored. IoT Hub gets the module distribution data, constructs these components, and verifies they are operating, for example by restarting the crashed components. IoT hub can also see how modules are doing. The IoT Edge cloud gateways is provided for the management of IoT devices. End-users could establish apps, transmit them to the device, and then keep tabs on how they're doing once they're there. Users may utilise this monitoring capability in large-scale deployment scenarios, where they can deploy programmes to a large number of devices and monitor their performance.

Using the IoT Edge interfaces, developers may create an IoT Edge modules image of their program and then upload it to their end devices using the Azure services or their own code. As soon as this data is acquired, a module image is downloaded and an instance of the module is created. A broad range of uses may be found with AIE. It now has use cases in smart manufacturing, irrigation, drone monitoring, and other areas. AIE is free source, while Azure service integration Azure function, Azure ML, and Azure stream, are paid for. This is worth highlighting.

CORD

A non-profit group called Open Networking Lab (ON.Lab) conceived of and launched the CORD Project in order to support open source ecosystems for the development of software-defined connectivity platforms and tools [6]. As a case study for the ONOS® open-source SDN operating system, CORD was born out of ON.Lab's discussions with AT&T regarding where the service providers' major expenses were. Central offices and the edge were examined by ON.Lab, which looked at how they were being constructed. When it came to establishing new services and competing with over-the-top competitors, we encountered a handful of limiting problems, such as a lack of adaptability. ON.Lab realised that in order for network operators to compete, the head office required to modernise.

As part of the CORD programme, AT&T built developed CORD as a use case in the context of ONOS. Service providers began to participate, concentrating on their goals in terms of the industry's move towards virtual networks. When ON.Lab's central office objective became a cross-industry endeavour, service providers, suppliers, and developers came together to support it. So many providers and resellers have embraced CORD in the last two years that, as of July 2016, it has been declared a stand-alone open-source project with an independent board of directors and a growing community of collaborators, collaborator-and-contributors actively working together to advance CORD development for use in production systems. The Open Networking Foundation (ONF) and ON.Lab are uniting to advance software-defined network and open source software technologies, such as CORD. ON.Lab collaborates with the Linux Foundation to guarantee that the project's governance is open, impartial, and merit-based.

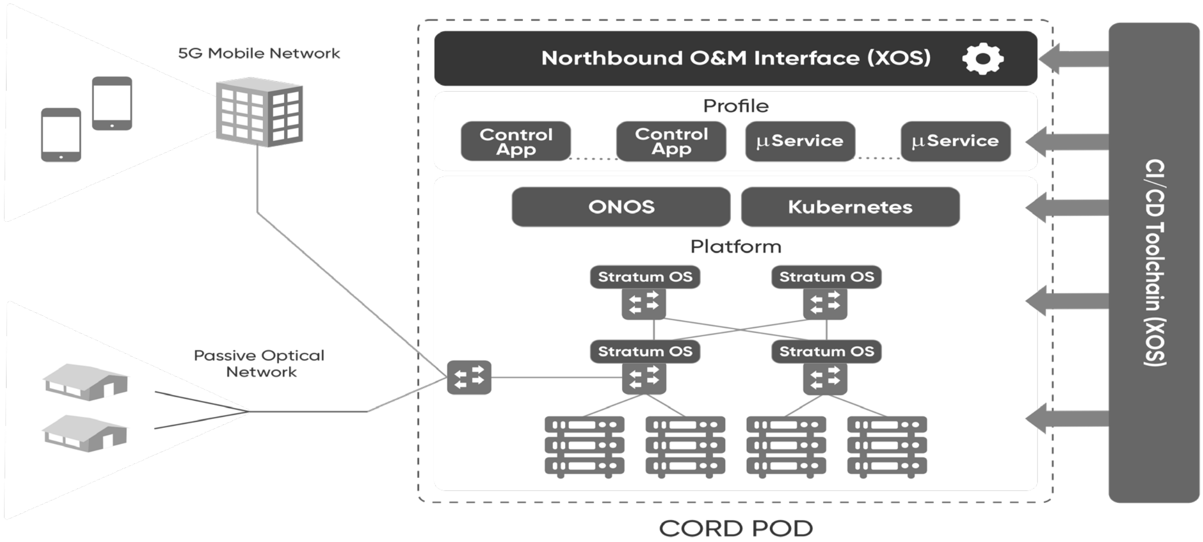

Fig 3. CORD hardware architecture

CORD is an ONF open-source project aimed at network service providers that was started by AT&T. (NSPs). Present system infrastructures are established on registered closed integrated schemes issued by network providers. The network is unable to dynamically raise or reduce its capacity because of its constrained nature. Computer and networking resources are wasted because of a lack of flexibility. SDN, NFV, and Cloud technologies will be utilized to establish information centers at the network edge. When these datacenters are used as cloud at the edge, they will be able to provide more responsive services to end users.

CORD is a combination of open-source software and commodity hardware. CORD's hardware architecture is shown in red in **Fig. 3**. White-box switches and common servers are utilised to connect the network. It is possible for an SDN controller to determine data flow through SDN switch employing white box switch. Commodity servers' processing and storage capacity are connected through switches to form a network. Spine-Leaf topology is used to create this switching fabric by adding horizontal system parallel to mainline longitudinal networking system, and then establishing analogous switching networking systems onto a horizontal network.

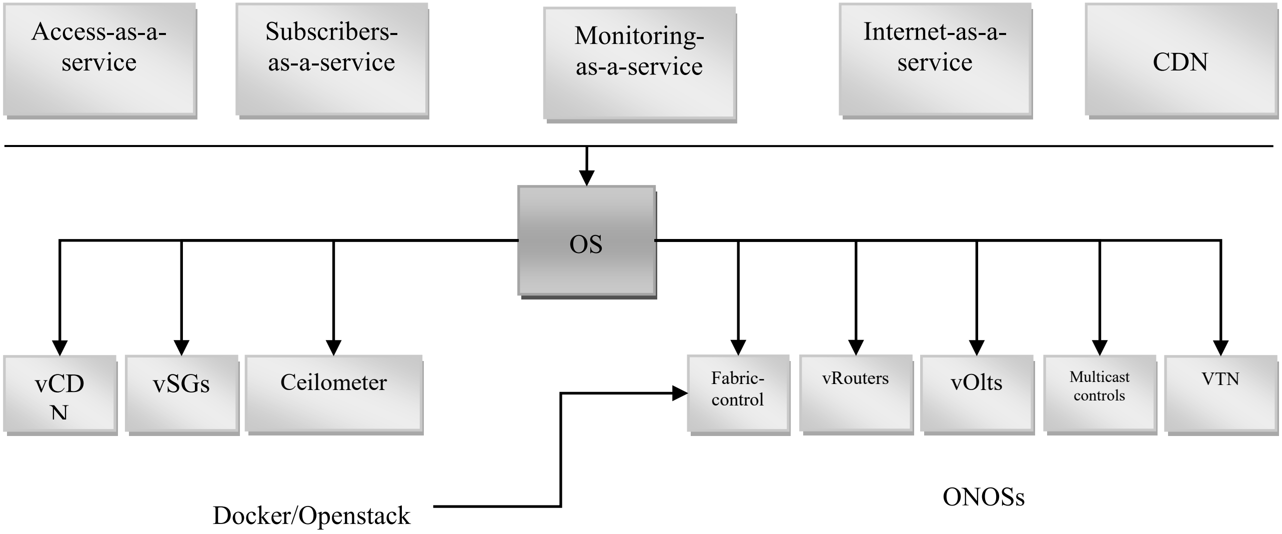

Fig 4. CORD software infrastructure

LAN traffic-oriented horizon-wise on network diagrams, such as those that represent traditional three-tier network architecture, enables scalable performance for bigger East-West networking traffic. Additionally, more specialized access tools are needed to interconnect end-users, mobile, business and residential users may all be subdivided into three categories. Each kind of access technology requires a different type of hardware. CORD's software architecture is seen in **Fig. 4**. Servers and switches in an OpenStack network provide IaaS capabilities for CORD. Controls compute, storage, and networking resources and generates virtual machines and networks, and virtualized networks and machine.

Dockers run each service in a distinct container, separating them from the rest of the system. It is an operating system that maintains networking elements, integrating switching fabrics, and distributed communication services to users. ONOS (Open Network Operating System) The control plane of XOS may be used to construct and compose services of various kinds. The vRouter (Virtual Router) project is an example of a software project that provides component capabilities.

When it comes to data, the operator's network's edge is a great place to start with the edge computing process. CORD has made the leap to enable edge computing in version 4.1 when it comes to offering edge cloud services. Mobile CORD, Residential CORD, or Enterprise CORD may be used in a number of ways, depending on the scenario, by implementing it as one of the three basic kinds of solutions. M-CORD that concentrates on mobile networking systems, mostly 5G networks, aims to decompose and virtualize the operations of the cellular network. Mobile applications may employ a variety of different edge services as a result of this versatility. In order to run your edge applications on a cloud-based platform, you may rent the service. E-CORD and R-CORD, they are established to be agile systems for the delivery of services, and for commercial or residential users. CORD is still being tested by the operators of networks, and more analysis is required to assess how it can interact with different edge programs.

Akraino Edge Stack (AES)

Akraino collection of open infrastructures and application blueprints, which can be utilised by both service providers and corporations, may benefit edge applications like 5G, AI, IaaS/PaaS, and IoT. These plans were created by Akraino users and focus only on the edge in all of its forms. All of these edge plans have one thing in common: they have been carefully tested by the community and can either be utilised as-is or customised. All-encompassing peripheral cloud infrastructure solutions are the objective of the Linux Fundament program AES, which was created and is currently operated by AT&T. Network operators are developing an open-source software architecture in order to satisfy the computing goals of zero latency, better accuracy, uptime, and adaptability.

Full solutions from architecture to application level are provided by AES. A three-layer scope may be shown in **Fig. 5**. At the application layer, where cutting-edge applications reside, VNF ecosystems are in demand. The second layer is the middleware that allows the top-layer applications. Third-party Edge applications like EdgeX Foundry will be able to use Akraino's API and infrastructure at this layer. AES is building an open-source edge architecture application architecture with the cooperation of upstream groups. Other open-source programs, such as Kubernetes, might easily be incorporated into it. ' As a prescriptive setup of hardware, program, and the delivery center, AES provides several edge applications instances. Plans for the IoT in Telco and other sectors, such as enterprise and industry, began with these plans, which are expected to be implemented in those areas in the future.

Examples include AES's announcement of Edge Media and Micro-MEC Handling as upcoming technologies to be developed. For smart urban, MEC's micro-goal is to construct new network infrastructures that allows for the development of smart buildings and a big data capacity for people. Edge Multimedia Handling intends to establish a cloud-based infrastructure for real media synthesis and AI insights at the network's interface. By creating a new smart urban service infrastructure, Micro-MEC promises to make it possible to supply services and offer people with a massive amount of data preservation.

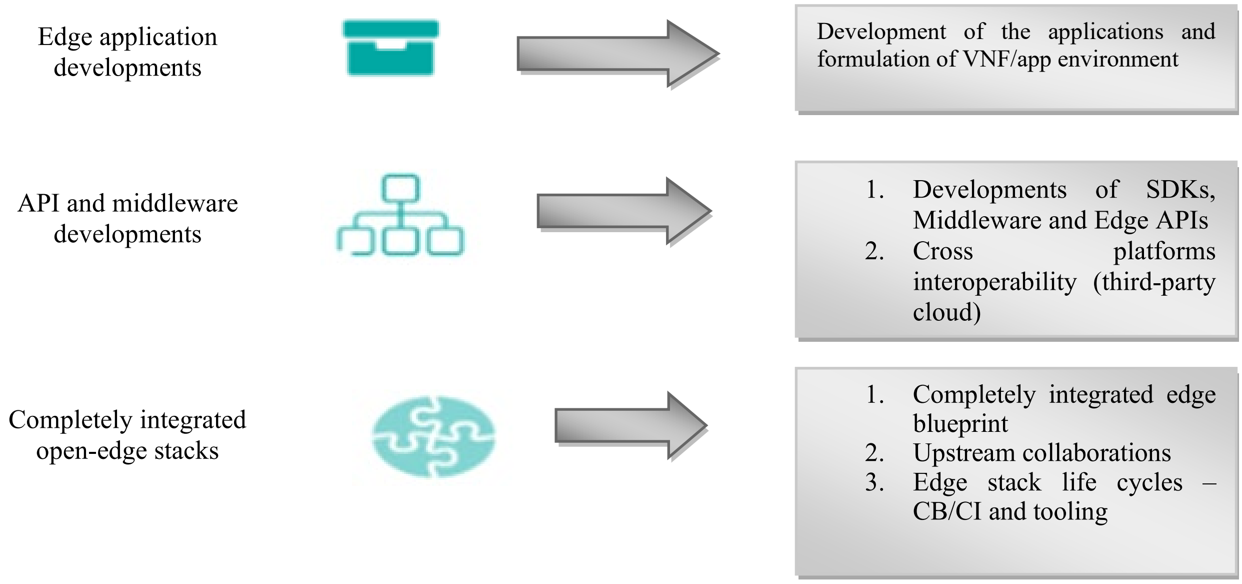

Fig 5. The scope of the AES

If you want to process real-time video with minimal latency, you need a network cloud like Edge Media Computing. Efforts to implement the AES have been ongoing since August 2018. With this project's advancement, further study is needed.

EdgeX Foundry

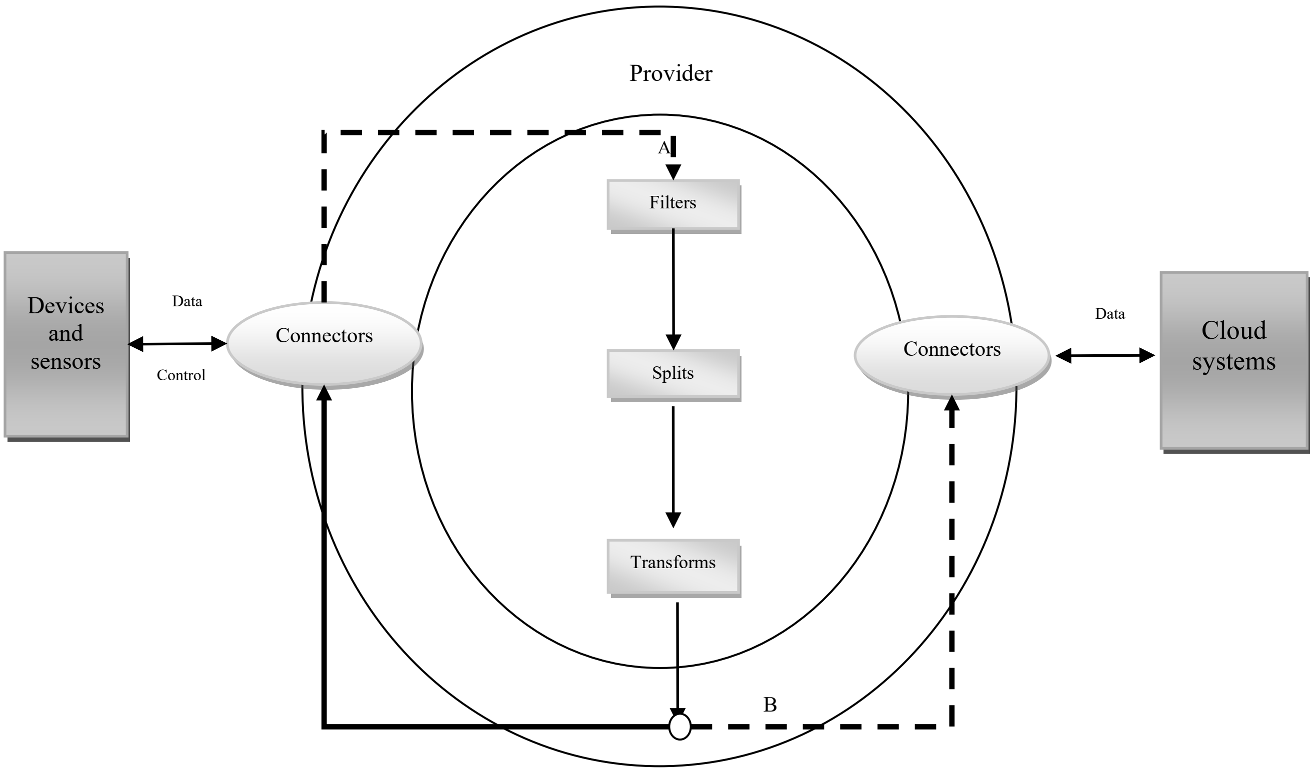

Fig 6. Edgent applications of the edge device

A common interoperability framework for the edge node e.g., routers, hubs, and gateways, is provided by EdgeX Foundry [7] for IoT edge computing. Data may be transferred to a local or cloud-based application for further processing from a range of sensors and devices that can be linked through a variety of protocols. To operate EdgeX, you don't have to have a certain CPU or operating system. As an alternative, a docker container or a native process may be used. According to **Fig. 6**, EdgeX Foundry's architecture may be noticed.

All of the IoT devices, sensors, actuators, and other IoT things are shown on the image's "south side," which links directly to the network edge. It's up to the "north side" of this diagram, which includes the network connecting Cloud, to acquire, aggregate, process, and convert data from Cloud (or Enterprise system). EdgeX Foundry connects everything, regardless of the hardware, software, or network you're using. EdgeX is working to standardise the way south-side IoT devices may be managed by north-side apps when it comes to IoT. EdgeX utilizes device profiled to illustrate the south side objects.

In this section, we will discuss about the kind of object, what data it may provide, how EdgeX will store that data, and the commands you can use to interact with it. In order for IoT devices to comprehend the commands, there must be a Device Service for each Device Profile. EdgeX provides an SDK for developers to build DeviceServices that may be used with any combination of devices and sensors as shown in **Fig. 6**. EdgeX's microservice design allows services to be scaled up or down depending on the capacity of the EdgeX-enabled devices. The micro-services could be distinguished into 4 distinct service tiers and two augmented system's layers. System Management and Security are the two basic enhancing system services that make up the four service levels that make up the four service layers. Using a single Restful API, each of the six layers is made up of a number of components.

The first layer is Device Services. Converting and transmitting data to Core Services, and subsequently interpreting instructions received from Core Services, are both responsibilities of the Device Service Layer in the Device Profiles. Core Data, Command, Metadata & Registry, and Registry & Configuration are all part of this layer. As a data storage and management platform, Core Data is a great choice. This is where object data from the south side is stored and handled. Command offers an API for users on the north side of the network to use when requesting commands from the south. There is no difference between a meta-data repository and a management service in the Internet of Things. A user's devices' metadata, for example, may be stored in metadata. Registry & Configuation provides centralised control of other microservices' settings and operating parameters.

The third layer of support is designed in order to provide cutting-edge analytics and intelligence. All the microservices that make up the Rules Engine have now been implemented to their fullest extent. By monitoring incoming data, Rules Engine may aid in the actualization of an existing rule by activating the specified device when the desired data range occurs.

It is possible to alert another system or person of an urgent actuation, a service outage, or other event by email, REST callbacks, or other methods. In the scheduling module, a timer may be set up to clean out old data on a regular basis. Tracking EdgeX's progress is made possible via logging, which is used. The Export Services Layer, which comprises Client Registration and Export Distribution, is the conduit via which EdgeX communicates with North Side. Client Registration enables third-party services, such as specialised cloud services or local apps, to request data from Core Data. Customers enrolled in Client Registration will be able to view the data after it has been distributed through Export Distribution.

Since EdgeX is dynamically extensible and may be deployed dynamically, management methods include installation, upgrade, start-stop-monitoring, and monitoring. EdgeX Foundry's security protects IoT devices attached to EdgeX Foundry against unauthorised access and modification of their data and instructions. IoT To manage massive numbers of sensors or devices, EdgeX is designed for automated factories, mechanical systems and many other IoT-related applications. New features and capabilities will be added to EdgeX Foundry in the future, as the product continues to evolve rapidly. The EdgeX UI, a web-based tool for adding and managing devices, is presently under construction.

Apache Edgent

Previously identified as Apache Quarks, Apache Edgent is presently the incubator computing project. Lightweight data analytics are made possible by using this open source programming paradigm in network edge devices like routers and access points. A key emphasis of Apache Edgent is edge analytics in order to accelerate data analysis development. Edgent's API enables developers to design cutting-edge applications. Edgent uses a topology graph to describe data streams abstracted into a Tstream class. It is possible to receive real-world data streams from the devices and sensors, or to communicate actual-life data streams to the cloud or other back-end systems through a link. EDGentis' API for data analysis is the system's foundation. Streams of data may be used to execute a variety of topological computations. Using a provider as a factory, Edgent constructs and runs network topologies. For an Edgent application, you'll need to acquire an Edgent provider, establish topology, and apply the processing flow to effectively manage data streams. Java 7, Android, and Java 8 are all possibilities for installing Edgent.

A variety of back-end systems, including MQTT, IBM Watson IoT Platforms, Apache Kafka, and convention text hub implementations, may easily transfer data to Edgent's APIs. The data that Edgent applications capture and deliver to the back-end system is analysed by a back-end system. In IoT applications, Edgent lowers data transit costs and delivers local feedback. Edgent may, for example, be utilised in Internet of Things (IoT) applications like autonomous transportation and manufacturing. Edgent programmes may gather data other than sensor data, such as files or logs. Thus, Edgent is able to serve a wide range of purposes.. It is possible to inspect error logs without affecting network traffic if it is integrated into application servers.

III. COMPARISON ANALYSIS

Comparison Aspects

These OSEC are summarised in **Table 1**. **Table 1** includes comparisons based on a variety of criteria, such as the primary function of the technologies, application domain, deployments, target user, virtualization, system feature, restrictions, scalability, and portability. Comparing open - sourced edge computing technologies in this way helps us better understand their present state.

Primary Function of Technologies

Our ability to identify a system's core objective helps us evaluate which one is most suited to run edge applications. EdgeX Foundry is an IoT sensor and device interoperability platform. Edge applications that make use of data from several sensors and devices must have this functionality. New edge applications may be developed using cloud-like development processes or by migrating existing cloud apps to AIE. The ability of Apache Edgent to speed up data analysis is well suited to Internet of Things (IoT) application cases. CORD aims to construct datacenters that can provide flexible network services to end-users by re-engineering the current edge network architecture. From the view point of edge computing, CORD allows many users with multi-access edge computing service. Utilizing AES, a high-availability edge cloud architecture may be achieved.

Application Domain

Edge solutions for the Internet of Things (IoT) [8] have a special focus: EdgeX Foundry is targeted on sensor connection, while Edgent is more concerned with data processing and data storage. Sensors and equipment in smart manufacturing, transportation, and cities generate data on a continuous basis. The Azure cloud may be thought of as extending into the IoT Edge's network of devices. Because of this, it is heavily dependent on edge devices for its many functions. Machine learning and image recognition can be easily deployed on the AIE for cloud deployments using Azure services. No application area limits are placed on CORD or AES when it comes to providing edge cloud services. Both of these methods allow individuals to connect to the operator network and run resource-heavy and interactive programmes, even if their edge devices lack the computing power necessary.

Deployments

AIE and Apache Edgent are installed within the edge computing devices, e.g., router, gateway, converters, and so on.

Table 1. Comparison of open edge system features

Aspects	AESs	CORD	Apache Edgents	AIE	Edge 'x' foundry
Portability	Supportive	Supportive	Unsupportive	Unsupportive	Supportive
Scalability	Changeable	Changeable	Unchangeable	Changeable	Changeable
Restrictions	Incapable of being offline	Unable of being offline	Limits to big data analytics	Azure services are chargeable	Lack of programmable interfaces
System feature	Widespread edge cloud	Widespread edge cloud	API for big data analytics	Powerful Azure service	Common API for device management
Virtualization	Virtual machiine and containers	Virtual machiine and containers	JVM	Containers	Container
Target user	Network operators	Network operators	General user	General user	General user
Deployments	Dynamic	Dynamic	Static	Dynamic	Dynamic
Application domain	Unrestricted	Unrestricted	IoT	Unrestricted	IoT
Primary function of the technologies	Support edge cloud with open-source sofware stacks	Transform edge of operator networks into an agile service delivery platform	Accelarating the developments process of big data analytics	Support hybrid cloud-edge big data analytics	Providing interoperability for IoT edge

Users may self-deploy EdgeX Foundry, dynamically add or delete microservices, and customise the user interface in order to run their own edge applications. A cloud-based interface is required for AIE deployments and software development. As a result, the AES and CORD stacks are equipped with a variety of network devices, including fabric switches, wireless access devices, and network cards. Customers don't need to own and maintain their own servers; they may instead use the services provided by network providers as a cloud service. Because there are no hardware or management needs, clients don't have to worry about this.

Target user
Despite the fact that they focus on edge computing, these open-source platforms cater to a variety of users. Target users are unconstrained in EdgeX Foundry, Apache Edgent, and AIE. Anyone with access to a gateway, router, or hub may install them on the local edge. To cater to network operators, they are meant to focus on the edge infrastructure. They are two separate products.

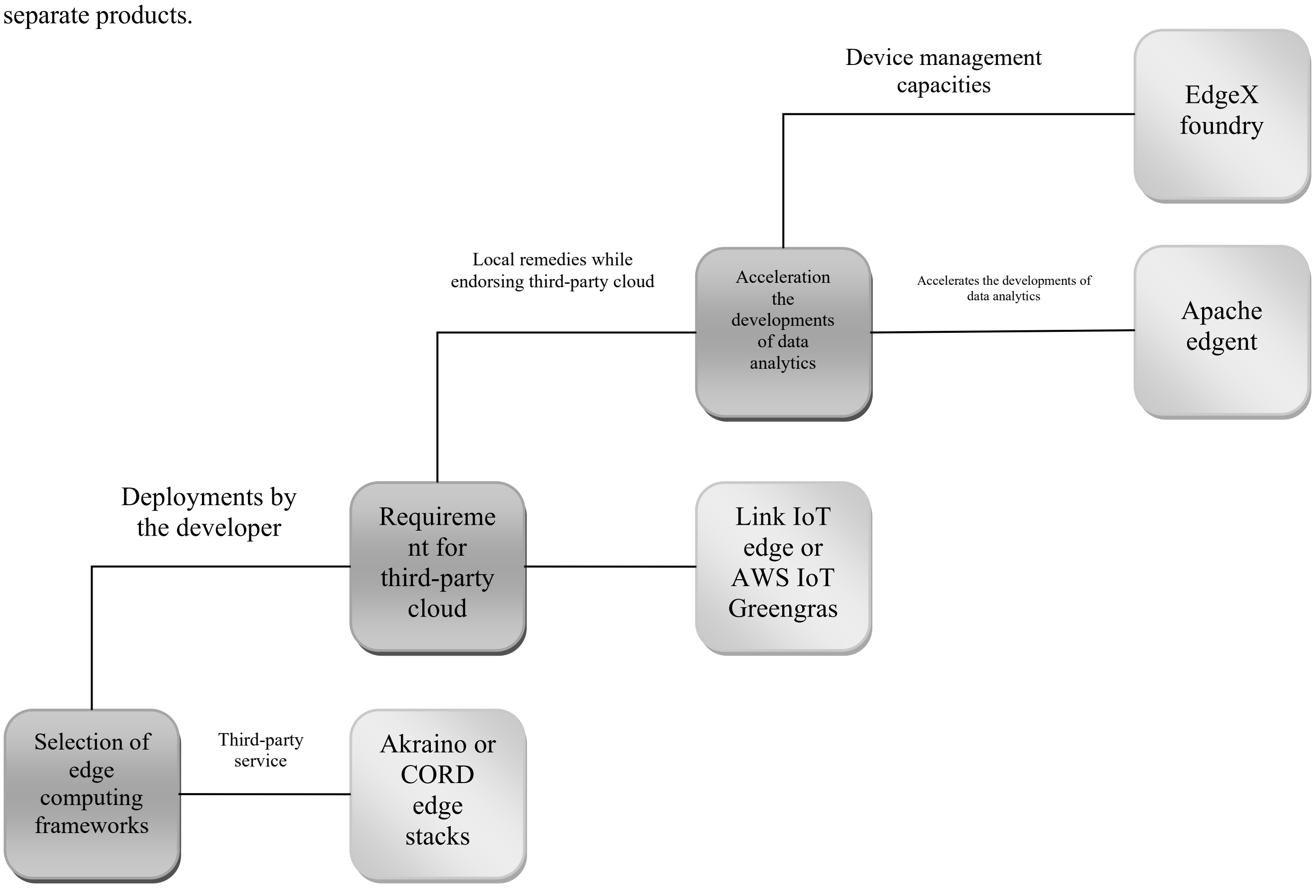

Fig 7. The selection of open-source equipment in various cases

Virtualization
In today's environment, virtualization is a frequent practise. A company's operations may benefit from virtual machines, which may help with resource use, stability, scalability, and other factors. A low-cost method of providing isolation and agility to devices in the periphery is container technology. OpenStack and Docker may be used in edge clouds thanks to CORD and AES. Different combinations of hardware and software may be found on the edges of the network. It's possible to run services on edge devices that are completely independent of the operating system they're running on. As a result of this modification, AIE and EdgeX Foundry now operate in Microservices. The JVM (Java Virtual Machine) is used to execute Edgent programmes.

System Features
Using a system's attributes may help in the creation, deployment, and monitoring of cutting-edge applications. I believe this functionality will save you a lot of time and work in the future. EdgeX Foundry provides a consistent API for large-scale deployment and monitoring of edge applications. To accelerate the development of edge apps, AIE delivers strong Azure features. Edge analytical apps may be rapidly and simply built using Apache Edgent's simple APIs. CORD and AES are used on the edge cloud to enable multiple access services. We do not even require an edge software platform on network edge to provide these operations, as long as we have an active web connection.

Restrictions
There are several limits to installing edge applications in the most recent version, which will be discussed below. In the most current version of EdgeX Foundry, developers are unable to construct their own applications. It takes a lot of time and work to incorporate unique features, even with EdgeX's versatility and power. However, even though AIE is free,

software application like Azure requires payment. Apache Edgent, on the other hand, is compact and concentrates only on big data analysis. In order for CORD and the AES that functions on the edges of operators' networks, to function properly, there has to be a strong network connecting data sources and operators.

Scalability

For network design, managing applications at the network edge is becoming more and more challenging. An important challenge in edge computing is its ability to grow. These edge computing systems all use docker or virtual machine technologies so that clients can simply add or remove module images so that they may scale up or down their applications easily. EdgeX Foundry enables customers to dynamically expand or reduce the number of microservices in order to better suit their individual needs in real-time. Because each Edgent application is an independent Java programme, Apache Edgent's performance cannot be dynamically adjusted, and considers accelerates the developments of data analytics as shown in **Fig. 7**.

Portability

Applications cannot be selectively transferred to other endpoints while running on edge nodes for EdgeX Foundry, AIE and Apache Edgent. In order to support 5G and 4G cellular services, AES and CORD are placed in the telecommunications infrastructures. These technologies are great for autonomous vehicles and drones because of their mobility.

Scenarios

On a Local Area Network (LAN) [9], IoT edge applications [10] can run without the need of cloud services, but they may also be used to connect to the cloud. Using EdgeX Foundry or Apache Edgent, clients may build and manage their own back-end systems, independent of any cloud platform. In order to monitor and handle a large number of edge devices, EdgeX Foundry is the best alternative. Consider utilising Apache Edgent instead of EdgeX Foundry for data analysis. Developers may swiftly build cutting-edge analytic applications that take use of a wide variety of data processing APIs using this framework.

In the second scenario, cloud services might be moved closer to the network edge in order to improve their quality. This is where Auzre IoT Edge comes in. It provides a simple way for developers to move their cloud-based applications out to edge devices and take use of high-value services and capabilities provided by third parties. Other candidates for the title of best competitive project in this field are Link IoTs Edge and AWS IoTs Greengrass from Amazon and Alibaba Cloud respectively. Microsoft AIE also offers Azure Functions, Azure ML and Azure Stream Analyzers as add-on services. It is possible to run cloud-based machine learning models on AWS IoT Greengrass once they've been created, trained, and optimised. In addition to Function Compute, Link IoT Edge offers a variety of additional functions. If an application's requirements are taken into account, it is feasible to choose the best system.

Suppose that third-party solutions are expected to be used directly by end users to install edge applications without the need for local hardware and/or software infrastructure in the third instance. It is recommended to use AES and CORD in this situation. There are a variety of options for users to pick from. Companies in the telecommunications industry rely on them. Communications firms also provide network service. It's possible that these technologies may be able to handle the specialised network requirements of edge applications. Since autonomous vehicles and drones rely on wireless telecom networks for their operation, using these systems' mobile edge computing services might be a viable choice for such applications (such as 4G or 5G). Additionally, there are several other open-source projects now under development. It is a component of ARM's Mbed IoT Platform, which provides access, administration, and control to edge devices. Huawei created KubeEdge, an open-source container orchestration platform for edge devices.

IV. CONCLUSION

A new concept of edge computing transfers the power of the cloud's communication, computation, and storage to the user's side all at once. Using edge computing in combination with the Internet of Things (IoT) and 5G seems to hold promise in terms of providing customers with enhanced services and applications. In general, new edge computing techniques and applications reduce data handling and communication overhead while improving the performance of mobile data analytics. Edge computing and machine learning are currently being utilised in tandem, which is furthering the area of edge-based intelligence services development. This article discussed several open source projects; it also demonstrated and made comparisons of the projects with key aspects e.g., primary function of the technologies, application domain, deployments, target user, virtualization, system feature, restrictions, scalability, and portability.

References

[1]. Linkedin.com. [Online]. Available: https://www.linkedin.com/in/rosaguntrip. [Accessed: 05-Mar-2022].

[2]. Cncf.io. [Online]. Available: https://www.cncf.io/wp-content/uploads/2020/11/CNCF_Survey_Report_2020.pdf. [Accessed: 05-Mar-2022].

[3]. W. Liang, Y. Ma, W. Xu, Z. Xu, X. Jia, and W. Zhou, "Request reliability augmentation with service function chain requirements in mobile edge computing," IEEE Trans. Mob. Comput., pp. 1–1, 2021.

[4]. "Worldwide spending on edge computing will reach $250 billion in 2024, according to a new IDC spending guide," IDC: The premier global market intelligence company. [Online]. Available: https://www.idc.com/getdoc.jsp?containerId=prUS46878020. [Accessed: 05-Mar-2022].

[5]. O. Ali and M. K. Ishak, "Bringing intelligence to IoT Edge: Machine Learning based Smart City Image Classification using Microsoft Azure IoT and Custom Vision," J. Phys. Conf. Ser., vol. 1529, no. 4, p. 042076, 2020.
[6]. L. Peterson et al., "Central office re-architected as a data center," IEEE Commun. Mag., vol. 54, no. 10, pp. 96–101, 2016.
[7]. J. John, A. Ghosal, T. Margaria, and D. Pesch, "DSLs for model driven development of secure interoperable automation systems with EdgeX foundry," in 2021 Forum on specification & Design Languages (FDL), 2021.
[8]. S.-H. Lee, T. Yang, T.-S. Kim, and S. Park, "TTGN: Two-tier geographical networking for industrial internet of things with edge-based cognitive computing," IEEE Access, vol. 10, pp. 22238–22246, 2022.
[9]. L. Yang, "Data acquisition and transmission of laboratory local area network based on fuzzy DEMATEL algorithm," Wirel. netw., 2021.
[10]. K. Venkatachalam, P. Prabu, A. S. Alluhaidan, S. Hubálovský, and P. Trojovský, "Deep belief neural network for 5G diabetes monitoring in big data on edge IoT," Mob. Netw. Appl., 2022.