Journal for Re Attach Therapy and Developmental Diversities
eISSN: 2589-7799
2023 November; 6 9s (2):1913 - 1926

Unifying Edge and Cloud Computing: A Framework for Distributed AI and Real-Time Processing

Ravi Kumar Vankayalapati[1*], Lakshminarayana Reddy Kothapalli Sondinti[2], Srinivas Kalisetty[3], Shashikala Valiki[4]

[1*]Cloud AI ML Engineer, Equinix Dallas, ravikumar.vankayalapti.research@gmail.com, ORCID : 0009-0002-7090-9028
[2]Sr software engineer, bank, Dallas, lakshminarayana.k.s.se@gmail.com, ORCID: 0009-0003-2070-3213
[3],Integration and AI lead, Miracle Software Systems, srinivas.kalisetty.ic@gmail.com, ORCID: 0009-0006-0874-9616
[4]Research Assistant, shashikala.valiki.researcher@gmail.com, ORCID ID: 0009-0008-2853-668X

Abstract

Edge computing and cloud computing are two popular computing paradigms that have distinct application scenarios and are often regarded as mutually exclusive. However, they have their own advantages and are often used together to support innovative applications like intelligent transportation, smart homes, and video analytics in smart cities. As edge nodes operate remotely with limited computing resources, different from cloud computing, applying AI to distributed environments facilitates real-time processing and the exchange of information. In addition, the way that unifies edge and cloud computing makes it preferable for engineering robot teams.

In this paper, we propose a cloud-edge computing migration framework for merging AI at the edge with the real-time processing power of the cloud, a different distributed AI framework, necessary components and algorithms, and the primary drivers of the aforementioned solution from a systematic point of view. At the same time, we collect the latest progress and clarify why the mentioned environment has a huge impact on applications such as collaborative robotics and self-driving vehicles. Cloud computing is at the other end of the continuum from edge computing, operating through the web with its resources, applications, and services. Depending on general-purpose processing, highly available services can be activated in seconds with the load of a button.

Keywords: Edge Computing,Cloud Computing,Distributed AI,Real-Time Processing,AI Framework,Edge-Cloud Integration,Distributed Systems,Latency Reduction,Data Processing,IoT and AI,Decentralized AI,Edge Intelligence, Cloud-Edge Convergence, Scalable AI,Autonomous Systems.

1. Introduction

The global transition to the Internet of Things (IoT) has resulted in a significant transformation in computer science. Embedded systems now have capabilities that can only be found in sophisticated deskwork systems, such as servers and data centers. The rise of edge computing has been the outcome of this trend. Edge computing, which is closely associated with the pre-existing cloud computing paradigm, allows for a new way of complimenting the infrastructure established by the cloud computing paradigm with limited processing power or data storage capability. This framework may increase the efficiency of distributed AI running at the edge.

Real-time procedures have long lived at the heart of several systems that are crucial to our culture. However, as the world becomes more interconnected with cyber-physical systems that are significantly larger, the demand for increasingly higher compromises is obvious.

The incorporation of real-time processing capabilities into technology has been possible thanks to the advancement of both processors and their respective operating systems. This technology can be found across a wide range of sectors, including defense, communication, transportation, and the military. To answer the identified questions, each aspect of this paper is divided into a number of sections. Section 2 examines the related works from the literature, evaluating available designs and comparing them in an analytical way.

Journal for Re Attach Therapy and Developmental Diversities
eISSN: 2589-7799
2023 November; 6 9s (2):1913 - 1926

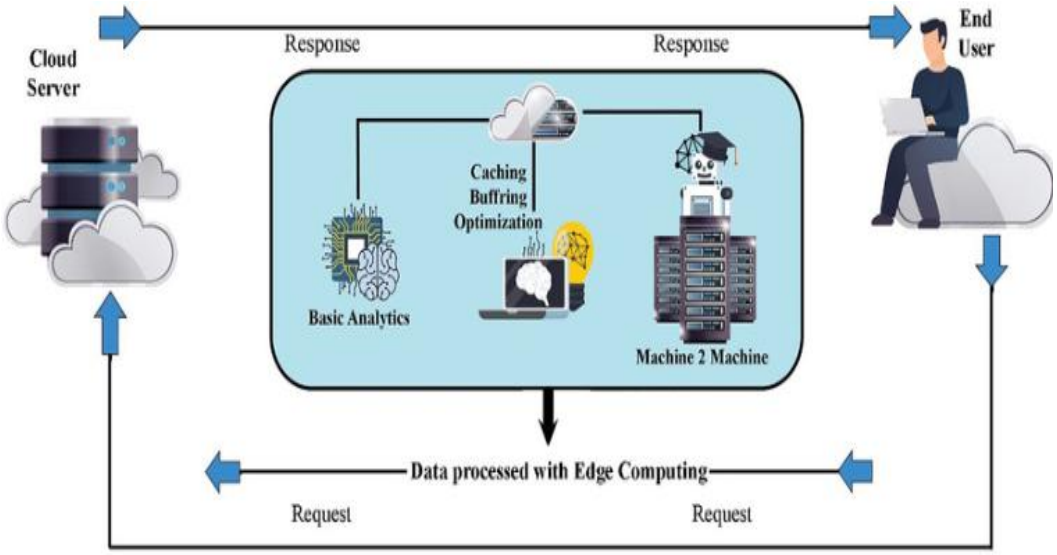

Fig 1: Transmission of data with edge computing

1.1. Background and Motivation

From its roots in centralized mainframes, computing systems steadily grew in capability to deliver low-cost and large-scale remote servers and systems now termed the cloud. In render farms and HPC clusters, these systems had significant computational power, but they were isolated systems that could not share capacity and did not provide a platform for general computation and application execution. Around the turn of the millennium, advances in technology led to these siloed systems being replaced by shared cloud computing platforms built on microservers, virtualization, and commodity open-source software stacks. Over the years to follow, other mainframe and HPC clusters started operating on a model of pooled resources, consumed in quantities to match the needs of specific enterprise and compute workloads. In recent years, a spectrum of real-time, sensory, IoT applications represents the convergence of computing and networking with physical-world phenomena. On the supply side, this is forecasted to drive unmitigated growth in data production. In industry, there are clear cases for low-latency ambient AI inference at the edge and many cases for data fusion and real-time alerting when there is a local, high-fidelity, and higher-temporal-bandwidth understanding of the phenomena. Therefore, today's cloud providers stand to gain from enabling a unified front at the boundary of cyber and physical worlds with more continuity, parallelism, and immediacy between remote and highly distributed resources through a common distributed systems and logic stack. This merger will also provide corporations across the edge-cloud continuum with far-reaching new means to exploit their end-to-end data assets with on-cloud analytics, deeply learned AI, and other real-time intervention technologies that operate across front-end edge outcomes and back-end cloud-level strategic and governance capabilities. Having matured this argument, edge, and cloud providers are now charged with developing and proliferating unified curves to this end, while effectively minimizing the unified edge-cloud processing steps needed to traverse between the cyber and physical worlds for optimal processing and response. Overall, this necessitates tight integration of edge and cloud capabilities, in particular, edge-based real-time analytics and cloud-based machine learning.

1.2. Research Objectives

This work represents an investigation into offloading computations, learning models, and data between edge and cloud environments, and how to integrate intelligence processing. To understand the needs and challenges faced by the two borders in their offloading decisions, we not only discuss traditional offloading aspects such as placement optimization strategies. We also focus on advances, in terms of both intelligence processing initiatives that look at embedding adaptations for offloading decisions, and data consistency issues, which have not been discussed from the perspective of the potential unification in a future smart distributed environment. Both offloading challenges underline the isolation and fragmentation caused by the decoupling of intelligence processing paradigms, which motivated the identification of synergistic opportunities between the two edges while considering the satisfaction of real-time processing requirements. In anticipation of any future offloading activities between the border edges, we discuss a large number of benefits in parallel across the infrastructure, including privacy, security, network capacity, and service quality.

Research Objectives: This study seeks to make significant contributions to the state of the art and knowledge on unifying the edge and cloud environments. Our corresponding research objectives include the following: 1) To justify the need for edge-cloud complementation using use case studies in the field of vehicular fog computing; 2) To develop a two-level decentralized deep reinforcement learning framework for decision-making in opportunistic service task offloading of vehicles in vehicular fog computing scenarios by combining several learning models; 3) To provide implementation and analysis of an adaptive offloading model in a simulated vehicular fog environment and conduct a sensitivity analysis for

performance evaluations; and 4) To experimentally validate the adaptability of the decision-making offloading model in two-vehicle driving scenarios.

1.3. Structure of the Paper

In the next section of our paper, we will first discuss some fundamental concepts that are closely associated with edge and cloud computing. Here, we focus on some key concepts that are essential to understanding edge computing, as well as their relevance in the present time. We will discuss how edge computing is an evolution of cloud computing. Further, we will offer a classification of edge computing, followed by a discussion on why we need to harness edge computing. We will then look at the necessary components that an edge computing system has to consist of. The third section will elucidate the essence of cloud computing, as well as its relation to the edge computing paradigm. We will offer a small conclusion at the end of this section.

In section four, we provide an overview of the integration of edge computing and AI. This section will serve as a useful foundation for understanding the more advanced material that follows. In the context of integration, we will first discuss the characteristics of AI. We will look at some of its basic definitions, circling back to discuss machine learning as a subset of AI. Here, we will present a discussion of distributed learning as the integration between AI and edge and cloud computing. Further, we will take a detailed look at the AI-driven data processing continuum for distributed AI. Section 5 will formalize the primary contributions of the paper. Further, we will go on to demonstrate a new methodology for real-time AI processing using an edge/cloud computing paradigm. This section will pave the way for discussion on future works that seek to undertake experimental validation with parameters that are grounded on real systems. The last section will reveal our conclusions. It will demonstrate how we can unify edge and cloud computing in the aforementioned manner and show the potential benefit of employing the described distributed learning in the proposed paradigm.

Equ 1: Real-Time Data Stream Processing

$$\dot{D}(t) = \lambda_{edge} \cdot D_{edge}(t) + \lambda_{cloud} \cdot D_{cloud}(t)$$

Where:

- $\dot{D}(t)$ is the rate of incoming data at time t.
- $D_{edge}(t)$ and $D_{cloud}(t)$ are the amounts of data processed by the edge and cloud, respectively, at time t.
- λ_{edge} and λ_{cloud} are the processing rates of the edge and cloud.

2. Fundamentals of Edge and Cloud Computing

Edge computing is a new paradigm-shifting cloud service towards the network edge. Its application domain includes contiguous devices and gateways responsible for data ingestion, preprocessing, and storage. Cloud computing, on the other hand, is based on a centralized infrastructure that provides services over the network. Cloud computing principles are high computation, storage, and high-speed networking facilities, and they are also far from the location of IoT functional devices. Edge computing characteristics include real-time data processing and leveraging geographical co-location of the service consumer and data generator. From the IoT and infrastructure perspective, the edge focuses on hierarchical infrastructure and applications to utilize edge-level support and cloud-based services. In a nutshell, edge computing provides for a single location or subset of locations, with real-time, deterministic, and low-latency computing, storage, and communication network services to provide intelligence and insights.

Essential differences between the cloud computing paradigm and its underlying mechanism for data access and analysis and edge computing expand the offline to low-latency real-time processing and feedback applications. This has led to differences in the design of the architecture itself. While cloud architecture is a centralized infrastructure where large-scale data centers are situated at far-off locations and provide minimal delay in information exchange, the essence of moving these large data centers closer to the edge of the network is called Edge Architecture. The emerging edge computing paradigm promises lower latency, scalability, agility, and geographical distribution but increases data saturation and security risks. The centralized application of cloud infrastructure makes edge computing an ideal partner for real-time analytics and machine learning applications. Although there are several benefits on the edge side, there are still challenges for edge-computing-based AI applications like security, resource management, programming models, and edge application architecture. These challenges provide cloud computing the opportunity to be a partner in the edge computing ecosystem. The AI model training and deployment at the edge site enhance machine learning capabilities, but many applications are running on the cloud due to large-scale data infrastructure.

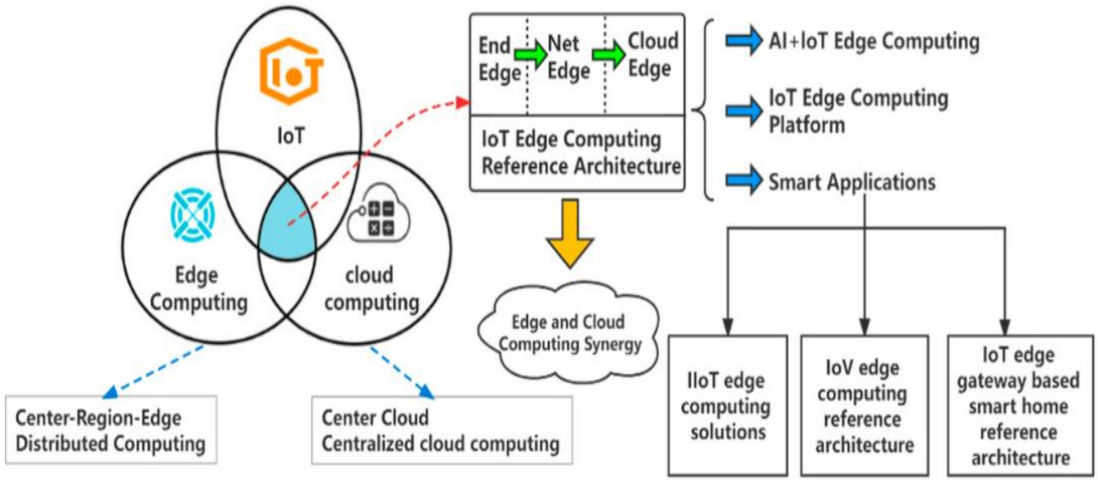

Fig 2: Fundamentals of Edge and Cloud Computing

2.1. Definition and Key Concepts

A core goal of distributed systems has been the ideal of incorporating information technology seamlessly into our lives. When we ask what terms like edge or cloud computing entail, we are approaching the definitions from a perspective that is based on the technology system, rather than the life systems we want to improve. As computers shrink in size and increase in power, we move applications and services from data centers into buildings, rooms, or even pockets. The terms edge and cloud computing refer to different approaches facilitating computing as close as possible to the final recipients of services, be these human beings or other machines. Technology is geared to fill gaps in our engagement with the web, spinning a faster, more convincing experience, and engrossing us in the result.

Edge computing, for the most advanced views of real progress in the distributed computing ecosystem, describes a place or a level of abstraction. These can be computational devices, networks, or virtual systems, such as containers, function as service environments, and the like. Edge gateway is a more specific term, describing systems referred to as 'edge' that act equally as a bridge between the edge alarming subspace and the cloud. This can be done by pre-processing data that is then sent to the cloud, for example, by browsing analytics on telemetry data, in order to reduce the volume of telemetries that are finally reported to the cloud. Edge devices are even more specific and specialize in data acquisition, connectivity, and rudimentary local processing and local alarming. These are to operate at the very edge of the infrastructure, including field devices or controllers, embedded devices, modules, and IoT hardware. These include Programmable Logic Controllers, Remote Telemetry Units, or wireless termini, depending on the remote interconnectivity mode.

2.2. Architectural Differences

Edge computing shares several conceptual similarities with centralized cloud computing but exhibits many valuable properties additionally. At an architectural level, cloud computing follows a centralized approach, while edge computing adopts a decentralized design. The former inherently employs a layered design so that different computation and storage layers exist, from the lowest to the highest layer. This is the essential reason why the amount of data generated and maintained is significant. Such a centralized design of cloud computing has multiple unique characteristics. One is that the placement of generated, processed, and stored data is on the cloud with high capability. The other is that the cloud is placed a large distance from the user. Since a large number of IoT devices and application users can generate a large amount of data, and because the cloud is placed at distances from the source, transmitting data over networks can degrade performance.

As such, network-based applications rely heavily on the upload and download of data between IoT devices, application users, and the cloud. Storing and processing a large amount of data on the cloud results in the consumption of huge amounts of network resources and energy. To address these limitations and attain advantages, the edge of the network has been used to manage and process data, which tends to be generated, received, and updated by users, wireless radio communication rigs, and devices in somewhat near proximity to users. In cloud computing, the location of the processing and data storage devices has significant importance. Interestingly, the location of data and processing sites is arbitrarily chosen in cloud computing, while in edge computing, the site is not arbitrarily chosen. In edge computing, data processing, and storage sites are positioned in a limited geographic area, i.e., closer to the user's point of interest. This is another significant aspect that distinguishes edge computing from cloud computing and has real-world implications. These architectural differences result in trade-offs in several areas, such as in performance, scalability, and resource utilization. In simple terms, the closeness of data processing to the data source and distance from centralized resources, or vice versa, affects the efficiency, cost, and architectural properties of the respective system. It also influences how applications are deployed in each architectural model.

Journal for Re Attach Therapy and Developmental Diversities
eISSN: 2589-7799
2023 November; 6 9s (2):1913 - 1926

2.3. Benefits and Challenges

Edge complementing and enhancing the cloud approach. Edge computing simplifies the data workflow and pushes computations done in the cloud closer to the source. Only a small percentage of the collected data can be of immediate value. Analyzing data locally provides faster response times, reduces latency, saves mobile bandwidth, and offers a better user experience by taking computational assets closer to where they are needed. This hybrid approach, where the two paradigms work complementarity, is known as edge computing. These optimization properties have clear advantages that make it a useful approach in various industry applications such as smart meters, grid protection, grid optimization, user appliance optimization, electric vehicles, power utility from production to consumption, e-health, smart manufacturing, or seismology.

Partnering this with the cloud gives the capability to manage the more localized distributed resources globally and perform computationally intensive operations. It eliminates the challenges in computing such as processing power, data storage, and software maintenance, and reduces network bandwidth. Each paradigm, the cloud or the edge, introduces its own benefits of deploying, managing, and utilizing resources. However, both paradigms also face similar challenges compared to the benefits they provide. In the cloud, these challenges could again be attributed to the size of the cloud platforms, and a lot of investment is poured in to alleviating the issues of security, data privacy, governance, and instilling customer trust, adoption, and interoperability. In edge computing, issues of equipment identification and authentication, equipment interoperability, collaboration among edge centers alongside load sharing, resource pooling, and related edge data centers across the enterprise, application, and service-specific resource allocation, maintenance of standard data center SLAs, and issues of catering for intermittent long-term health monitoring, as well as emergencies, applying and enforcing an acceptable use policy for multiple data-use scenarios with so-called shadow copies of its data. Edge computing calls for collaboration between edge and cloud in alleviating the lack of bandwidth between edge centers by using cloud resources and also exploring whether it is possible to somehow push some of the edge functionality and operations to the cloud.

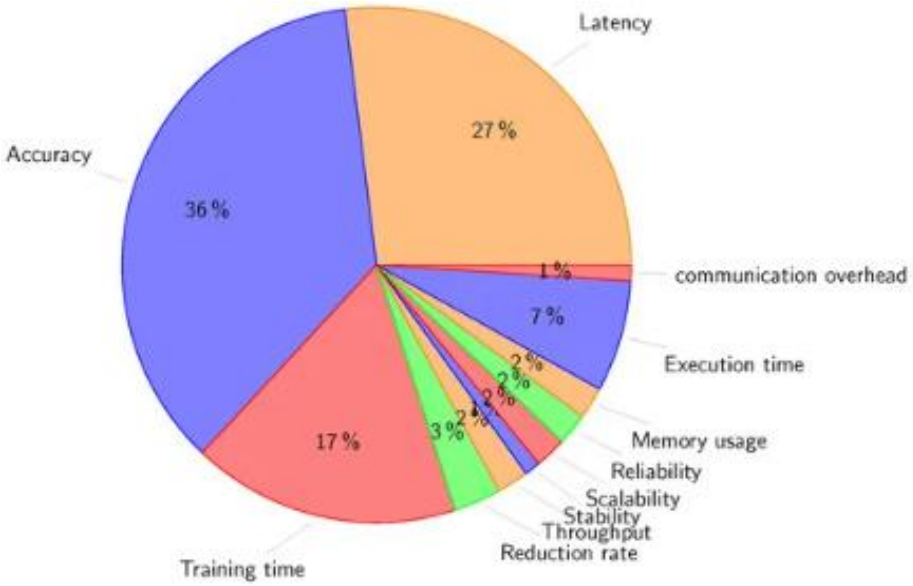

Fig : A Review and New Perspectives

3. Distributed AI and Real-Time Processing

We focus on three main directions in the current context of computing and AI systems, namely, (1) distributed AI, (2) real-time processing, and (3) relevant challenges of edge and cloud computing, especially those that bear the highest importance in designing appropriate frameworks for AI systems. The popularity of AI and machine learning in both industries and research communities has grown by leaps and bounds in recent years. Such AI-driven data processing needs to be carried out at two levels: on the one hand are edge devices, such as IoT devices, and on the other hand are cloud environments. This is already quite clear for today's applications because they produce much more data than those that can be handled by meshes of IoT devices alone. The amount of data produced depends on the use case and therefore introduces variation. This calls for systems that can handle both small-scale edges and cloud environments.

In many use cases, not all of this data needs to be sent to the cloud and can therefore be filtered, ranked, or handled locally and sometimes in a semi-distributed manner, meaning that some pre-processing might be needed before data can be aggregated between edges. Such systems often employ real-time analytics and handling of incoming streams of data. Real-time processing should be understood in this paper in both a strict sense, where any interaction with a computing

system happens in a rather well-defined amount of time, and a less restrictive sense of applications that have short-term reaction times. This includes but is not limited to, designing infrastructure-independent processing algorithms. When scaling is done through edge infrastructure rather than the specific deployment infrastructure, there are some new and profound implications for the way we deploy such systems. Chiefly, systems should be designed to disentangle learning from deployment specifications, which suits the general use of edge and cloud computing.

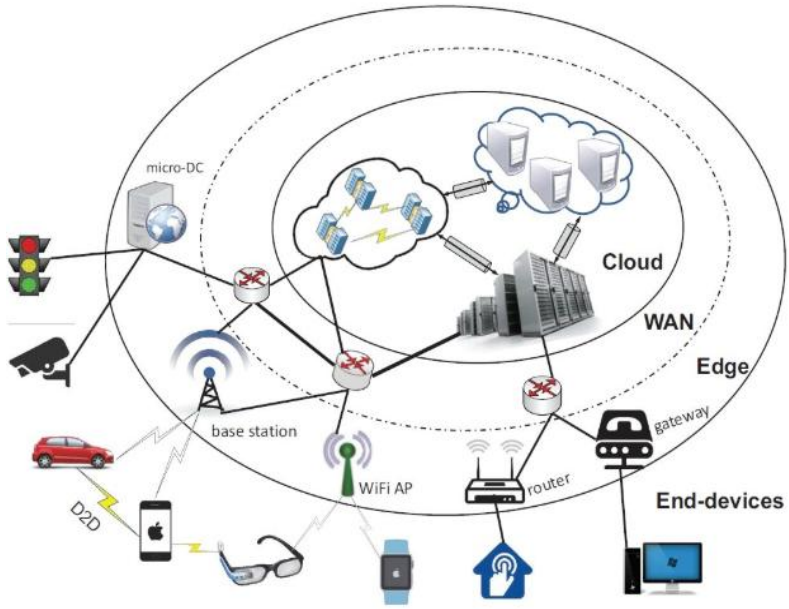

Fig 3: AI and Real-Time Processing

3.1. Overview of AI at the Edge and in the Cloud

AI at the edge level, also referred to as Edge AI, and at the cloud level, often termed Cloud AI, operate in different, yet sometimes interconnected, ways. The computational capabilities in both of these paradigms are significantly different and exist for different functionalities and operational advantages over one another. This has long been a motivation for researchers and practitioners to utilize AI at the edge owing to its ability to make real-time decisions based solely on local data and context. In contrast, Cloud AI operates by gathering large amounts of data from various sources and enforces multiple ML/DL models for obtaining in-depth data analysis.

In contrast to edge devices, which cater to relatively simpler data processing tasks in real-time, cloud-level AI utilizes broader and denser datasets to provide enriched services to the end users. Additionally, services based on edge and cloud AI are usually interdependent—both have their own degree of AI operations, which can differ from task to task. Consequently, future AI applications will likely see the implementation of both Edge and Cloud AI paradigms, which will function in unison to achieve collective objectives. This interdependence indicates the significance of context-aware computing in AI systems for achieving optimal user experience. In addition, integrating AI functionalities available at the edge and cloud AI levels will afford largely dependent systems that can provide an efficient end-to-end system. It must be noted that a larger part of the machine learning operations is also distributed to edge devices—lower layers of a DNN can be offloaded to edge devices—providing cloud-like functioning.

To that effect, the offloading or distribution can vary based on particular applications. Furthermore, the requisite for in-line processing features could also determine the localization of AI operations. It is widely accepted and proven that a completely local decision-making approach—including the AI stack—has its rewards and often only the system locality may be appreciable. Some of the AI paradigms are particularly designed for lawful enforcement in an edge-only scenario, such as on-board deep learning accelerators or kernels on inference-side GPUs.

3.2. Real-Time Processing in Edge Computing

Real-time processing is an important mechanism in edge computing as it allows processed sensory data to be immediately used for system decision-making. By making data processing closer to data sources, edge computing has the capacity to enable real-time processing. It is difficult for cloud computing and traditional AI solutions to provide real-time processing unless a large number of cloud servers can be available in the edge area. For the application scenarios requiring low latency, message time consumption is very important, while edge-based decisions close to the data source in private networks are also important in some industrial IoT applications, such as energy management and HVAC. As we see, many edge processing frameworks are primarily designed for the most cutting-edge mobile games and virtual reality experiences, focused on delivering massive quantities of high-quality content with ultra-low end-to-end latency. Time-

critical frame rendering with low surface submission latency is carefully calculated to keep the phone's display updated with the last rendered output.

However, it is difficult to guarantee real-time processing at the edge, as the edge devices do not offer powerful hardware and powerful GPU devices as in the cloud data center. Edge devices generally use low-power CPUs, lower memory, and slow I/O as well as limited storage. Moreover, the connection between the cloud and the edge is not always the best and fastest. In the cloud-assisted edge, connection loss or any change in the networks can affect edge systems. To address these challenges, some innovative and very low-power processing models have been recently presented to enable low-memory, real-time image vision applications at the edge. Further, some processors can be used for very low calling latency because of the on-chip memory processing capability. However, developing an expert system with such low computational capabilities is a challenging and complicated research area. Certain assumptions about the knowledge required by these models may limit their applications in various domains.

3.3. Challenges and Opportunities

The proposed solution poses several challenges and opportunities. First, there are clear interoperability and scalability concerns that must be addressed: workflows need to be distributed without having to rewrite an entire architecture. There is no silver-bullet answer to where information should be sent for computation, but it is clear that intermediate processing will need to happen at a few stages. In addition, data privacy concerns related to regulations restrict how data may be moved between edge and cloud. However, addressing these challenges is essential for AI to be deployed seamlessly in a way that is transparent to its users. Despite the challenges and some resistance from the public, there is no question that emerging technologies are becoming an intrinsic part of our everyday infrastructure and applications, and represent a great opportunity. For example, integrating distributed AI systems in edge/cloud environments could make traffic data available to optimize the delivery of goods and services, avoiding the economic and environmental cost of unnecessary congestion.

Studies investigating investments have confirmed that the adoption of connected infrastructure that relies on edge commoditized computing is evolving rapidly; estimated outcomes are in excess of €20 billion in revenue by 2022, with particular economic benefits due to improved health and well-being. Although there is a conclusive need and willingness from the systems' end-users to be part of this new distributed algorithm experiment, there are significant technical difficulties. Yet, despite the increasing demands and unlikely commercial value, investment is being made in new technologies to resolve identified technical barriers. Our goal in what follows is to address these technical difficulties and develop a middleware platform that can provide parallelism using as many of the display border devices as possible and, importantly, is able to exploit offload paths for intra-camera parallelism for nodes equipped with the right hardware capabilities. Research on distributed AI often focuses on the use cases and application domains, while work on real-time processing is always case-specific and there is no generic approach. We argue that there is a necessity for unifying technologies and approaches urgently in Edge. Finally, this text also covers technologies that are generic and more challenging in Edge; introducing concurrent systems, protocols, and services combined with integrating the infrastructure into open and shared cloud-based services. Given our analysis of the challenges and opportunities, we argue that research should consider AI systems distributed across the cloud and Edge. We will articulate the vision for edge services in due course.

Equ 2: Resource Allocation for Distributed AI

- α be the proportion of resources allocated to the edge, and $1 - \alpha$ be the proportion allocated to the cloud.
- T_{AI_edge} and T_{AI_cloud} represent the computation time for AI tasks on the edge and cloud.

The resource allocation equation can be formulated as:

$$\alpha = \frac{T_{AI\_edge}}{T_{AI\_edge} + T_{AI\_cloud}}$$

4. Integration Frameworks

Integration frameworks are pivotal to transitioning from the current state of distributed and scalable computing infrastructures to truly unified edge and cloud computing paradigms. Such integration frameworks are needed to deal with the unique characteristics of edge and cloud environments, as seen by a myriad of integration methodologies, such as distributed datasets and model synchronization. These technologies attempt to create a large-scale artificial intelligence

framework covering rapid real-time processing, as well as batch computation at the edge and cloud. In order to have true interoperability and end-to-end performance improvements, the method of integrating edge and cloud must be able to break the runtime up into different computation units guaranteed to comply with the tight timing constraints of distributed real-time applications and evaluate whether there are sufficient resources to process applications in real-time.

The final substantial hurdle for unifying distributed and scalable edge and cloud computing is their inability to work together. Interoperability at this degree cannot currently be achieved, very similar to how batch systems are natively incompatible with safety-critical distributed real-time applications. Current methodologies for enabling this unification are based on industry trends or made for specific applications of the technology and do not take into account the failure patterns and need for real-time guarantees of the edge. The current development environment is trending towards having an ever-increasing amount of hardware connected to the network, allowing researchers the luxury to partition workload data flows to work for individual devices. As such, researchers increasingly need to have demonstrated architectures and integration strategies to affordably eliminate data connectivity and processing friction to realize deployment to broader markets.

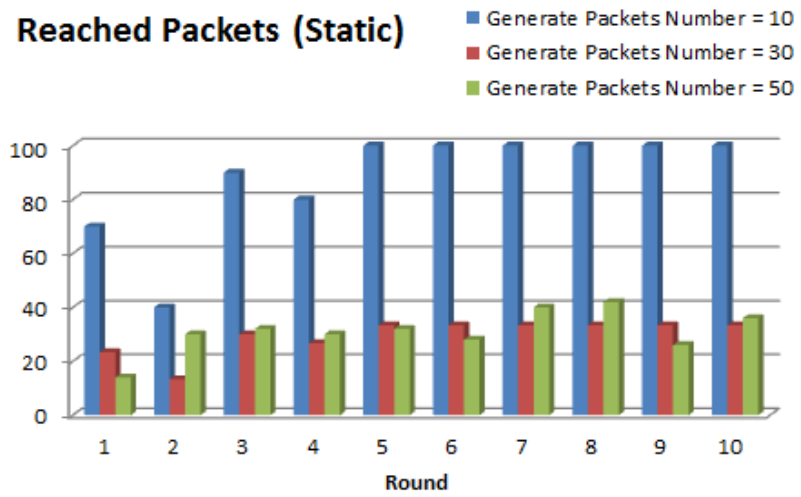

Fig : Percentage of reached packets sent based on the round number in both static and mobile environments.

4.1. Existing Integration Approaches

Designed to bridge the gap between edge and cloud computing, integration approaches have gained significant interest in both industry and academia. In this context, several strategies can be found, with the particular design often limited by the goals and partner organizations involved within individual projects or demonstrators. For example, support for object detection, video streaming, decision-making, training capabilities, and language design aids were among the numerous features supported by the proposed framework. Similarly, a range of approaches was highlighted, as were additional reviews on related topics. Indeed, it is evident that a number of similar concepts are available, but also that few implementations are available that actually follow through and consider a platform for mixed edge and cloud processing, without being application-centric or application-limited.

To align with the current position of edge and cloud computing, a description of the existing integration approaches that tie these two areas together is provided. A range of vertical integration approaches is employed, including both platform-based and application-centric models. A detailed analysis of related work in this section provides insight into which approaches provide the most complete output, with distinct strengths and weaknesses affecting real-world scalability in terms of parallel processing for a distributed system. By gaining an understanding of these systems, existing challenges may be fed directly into our integration framework. Subsequently, useful lessons can be learned in relation to the design choices made in each approach, where integration was a primary or secondary goal. It is important to capitalize on the insights from existing integration approaches, utilizing this understanding to design a framework based on the three main drawbacks seen in the literature. Thus, designing a scalable, streaming-based, real-time edge-cloud AI system became a focus of our research activities.

Journal for Re Attach Therapy and Developmental Diversities
eISSN: 2589-7799
2023 November; 6 9s (2):1913 - 1926

Fig 4: Existing Integration Approaches

4.2. Proposed Framework Design

This subsection outlines the design of the framework and our main contribution to middleware-level integration of the edge and cloud by solving problems. We summarize the previous sections' findings and investigate their faults. As a response to these problems, we introduce our design at the end of Section 4. This section is aimed at understanding the design of our integration solution, and the section thereafter illustrates experiments for proving the effectiveness of our proposed design.

This subsection introduces our proposed framework, an overview of which is given. We now detail the design of this framework. Our solution is based on the following observations and problems that are still open issues in the field. We foresee several challenges associated with our proposed framework. The proposed modular framework design decomposes disparate integrated components to provide remedies that fit the solution to the task. As a result, our approach is highly adaptable and scalable in comparison with other integration solutions. Moreover, our design is capable of real-time processing at the edge while maximizing resource efficiency.

Furthermore, the applications of various use cases describe a wide range of requirements. For instance, AI services distributed to the network edge require low latency and a limited network load. On the other hand, resource use should be optimized while dealing with cloudlet infrastructure. We can integrate the performance and universal issues by abstracting the architecture of our middleware level for edge-cloud integration. Consequently, it may be used for numerous future studies depending on the perspective.

4.3. Key Components and Interactions

Next, the following subsections further detail the integration of AI across system layers. Subsections include an outline of design principles, key system components, and the interactions between them, and finally, a high-level explanation of what happens inside the system when an AI component issues a control decision or sequence of control signals.

In this document, we describe a unified edge-cloud architecture called BUDDHI and present the flow from the cyber-physical environment through each of the key components of the BUDDHI architecture. We also make the case for why certain technological choices have been made. However, before we continue, we briefly present this architecture. At a high level, we describe several key components of the BUDDHI ontology and architecture, including the BFS that maps between BO and DD, the Ontology Reasoner (OR) that can update ontologies based on incoming data, the EN that interfaces with remote or local communications infrastructure to bring data from the outside world into the BUDDHI framework, and the Cognitive Engine (CE) that uses BO to infer actions that are then sent to DD to be executed. In this chapter, we also describe the integration of AI across these layers of the system function, where the realistic nature of the execution of those actions may require AI and ML to be used, at both the Digital Domain and the Cognitive Domain.

This section first identifies key components of the BUDDHI architecture before detailing how such components interact. For each BUDDHI component, the architecture is uniform, ensuring that components can communicate in a standardized manner. Interconnected components allow for the free flow of coded algorithmic information, which can all occur in real time due to the functionality of the edge, as outlined later. Latency is minimized through design, allowing for data to be shared between components through both cloud and edge when necessary. These design principles are essential in order

to create a system that can admit and process large amounts of data, serve multiple stakeholders from different domains, offer insights, and make decisions in real-time and near real-time.

5. Case Studies and Applications

Case Study 1: Fire Detection and Surveillance in Harbours. In this case study, we develop an integrated edge and cloud computing framework for the monitoring and surveillance of multi-camera unmanned ports. For this application, we used three cameras with an edge device. The results show that our method achieved low response latency with a high level of accuracy.
Case Study 2: Real-Time Health Monitoring. To evaluate the real-time capability of the edge and cloud computing framework towards health monitoring systems for elderly people, we implemented a two-stage health monitoring system. We used two largely pre-trained CNN networks, mainly pre-trained with a large dataset and slightly trained on our health dataset. We performed experiments to evaluate the two proposed stages on both a cloud server and an edge device.
Case Study 3: Real-Time Road Condition Assessment. This case study explores the application of the edge and cloud computing framework in road condition assessments. Two application scenarios were the focus of the case study. Scenario 1 addressed the development of a generic system for acquiring real-time information on road traffic conditions, while Scenario 2 focused on dual systems: Integrated Collision Alert and Road Condition Assessment. Our system involved employing an AI module integrated both with the edge and cloud framework servers. Moreover, we have used a CCTV camera with an AI edge device and a web-based workflow user interface. We have employed volunteers to collaborate on using vehicle resources as moving sensors to collect road vehicle-generated data, such as vehicle onboard camera video data. In this case study, our system has successfully generated the required operational real-time results under result validation. The risk of fire detection in these locations is significant. In addition, the fact that most of these cameras generate low-resolution images and have an old mainframe exacerbates the process of predicting this object. We verified the generic implementation of the diagnosed argument and contrasted this form of insertion in a hospital and an airport. Throughout this case study, a parallel server was used, which reduced the workload on the server used as an edge; this solution was adopted to overcome this problem.

Fig 5: Edge AI and Machine Learning Applications

5.1. Industry Use Cases

In this section, we present several industry use cases where the integration framework has been used to show the practicality of the approach. The use case describes an AI/ML framework for the edge and cloud used to alert the neurologist about stroke onset and related conditions. It includes a customized data layer and a decision that makes use of compressed signal classification and VAE, which aims to improve its computational performance at the edge by using distributed AI. The proposed framework has allowed us to provide real-time results based on two requirements provided by the Grove Memorial Centre we were working with: interpretability and trust. The overall system has been tested through 10,105 cases, demonstrating a similar performance from the bottleneck layer to the expected results by the neurologist, with no significant difference, guaranteeing its reliability and embedding distributed AI in the service delivery. We used the AIoTES architecture to highlight a use case in a smart city domain, which often distributes the information stored in the cloud and edge to cover as much physical space as possible. We participated in the 2nd AIoTES Plug Fest 2022, along with 15 other projects in the field, about edge, fog, cloud computing, and IoT, to show an AI

integration to bring distributed edge toward the future. We showed a monitoring system made up of different networks, composed of cloud and edge devices. We used AI in the cloud to assist the groups hosted on the edges.
The use case presents advanced analytics for a predictive maintenance system. The sensitive complexities in handling IoT data should be managed inside the edge to make it scalable and manageable. The proposed EDGE-PLUS system turns the edge into an advanced predictive analytic, whose effectiveness in terms of rapidity, accuracy, offloading, and dynamic actions are evaluated. When the system is implemented in each scenario, greater than 80% user satisfaction with the IoT data analytics is achieved, which is excellent for achieving the proposed system vision. The use of a distributed AI/ML model based on the edge and cloud at the same time is assessed. The prototyped distributed AI is evaluated based on the computation offloading time performance to the edge device and the keep-alive feature to maintain the distributed AI between edge and cloud models, ensuring reusability and better predictability. We have evaluated the overall performance of the cloud model at the edge, connected, and disconnected models in handling the percentage computation load, the time spent, the size of incoming data, and the accuracy of the resulting predictions. The identified key metrics include computation offloading time and the keep-alive feature for the winning model, percentage of computation load offloads, total time needed for handling incoming data, and accuracy between the true output and prediction results for user satisfaction.

5.2. Research Implementations

We have presented the theoretical foundation needed for designing a consistent, end-to-end Edge-to-Cloud computing framework built through a layered model. This subsection will cover how the theories we have presented have been implemented and evaluated by researchers. The goal is to present a larger context of research that converges with the fundamental concepts presented, thus validating our approach. One of the main outcomes I would like to also discuss here is the potential for creating commercial prototypes or pilot studies, thereby increasing the concreteness of the approach presented and increasing the impact of the approach. Some disjointed research efforts can be used as a reference from a practical standpoint. It is expected that future research can propose, as to the meta-model presented, commercial, technical, and more detailed implementations.
The discussion includes AI workloads for the Edge and Cloud. The design of a solution management paradigm for the Internet of Things (IoT) is intended to alleviate the challenges. To demonstrate the solution, the study presents three applications: home automation with detection and recognition, environmental monitoring, and healthcare. Continuing this, the participants analyze in detail the underlying training deployment. Last but not least, the research results are commented on. Just as important, the work highlights the impact on society and suggests possible points of collaboration between industry and academics on potential case studies. Designed to inform microservices, an intelligent policy management framework for graph-based computing from the network edge to the carrier cloud is presented and the results of the study are elaborated on.

5.3. Performance Evaluation

This subsection delves into the performance evaluation of the proposed integration framework on distributed AI applications at the edge and cloud. Both IT industry and communication use cases are presented to underline the conclusion.
To critically discuss the performance evaluation impact of the proposed IAICU, the following should be further discussed in this section:
The performance evaluation of the proposed IAICU is a vital process and is of great interest to different stakeholders. A sound and transparent evaluation is essential for different stakeholders to assess the suitability and acceptance rate of the proposed framework. To put the evaluation into perspective, the analytical method is addressed, as well as the performance metrics employed. Moreover, several practical parameters affected are analyzed, such as processing speed, energy consumption, latency, and resource utilization. Extensive studies have been conducted to evaluate the proposed framework under real scenarios, and results are both benchmarked against prevalent solutions and compared for both industry and academic evaluations.

As a result, several challenges face the evaluation process due to real target prototypes along with the developed use case testbeds, and the need for continuous monitoring with possible optimization and enhancement that points towards future research. Ultimately, the results obtained were used to clearly demonstrate the enhancements brought by the IAICU framework for both industry use cases.

Equ 3: Energy Consumption in Edge-Cloud Integration

$$E_{total} = E_{edge} + E_{cloud}$$

Where:

- E_{edge} is the energy consumed by the edge device to process its part of the computation.
- E_{cloud} is the energy consumed by the cloud servers to process the offloaded data.

6. Conclusion and Future Directions

In this paper, we show that unifying edge and cloud computing can have many benefits, especially from the perspective of distributed AI. In particular, we argue that disintermediation allows for unifying the best of both worlds, and current trends indicate that both cloud computing and edge computing will be important in the future. Using the lens of technical disintermediation, we established the following thesis: a tight integration of edge and cloud computing is required to accelerate the unrolling of massively deployed distributed AI and to perform real-time processing. Especially in new application scenarios, the ability to perform real-time processing is of paramount importance. In contrast to existing solutions where both fields are treated more or less as separate islands, a unification promises to de-silo edge and cloud resources and offer a continuum for latency-critical applications. We showed that, in addition to clear technical and scientific arguments, ongoing trends point towards this unification. We also discussed the open challenges and outlined how this vision can materialize in progressive steps. In future work, it is important to merge the ongoing research in edge computing with this new vision of unification. Topics in the area of data-centric edge architectures and edge AI can also find new directions. By unifying the data management services, we can rebuild the upper-layer services. This section provides a reflective closure to the paper by summarizing the main contributions made. Finally, it discusses the main future trends in technology that might influence this field as well as future avenues of work from the research community.

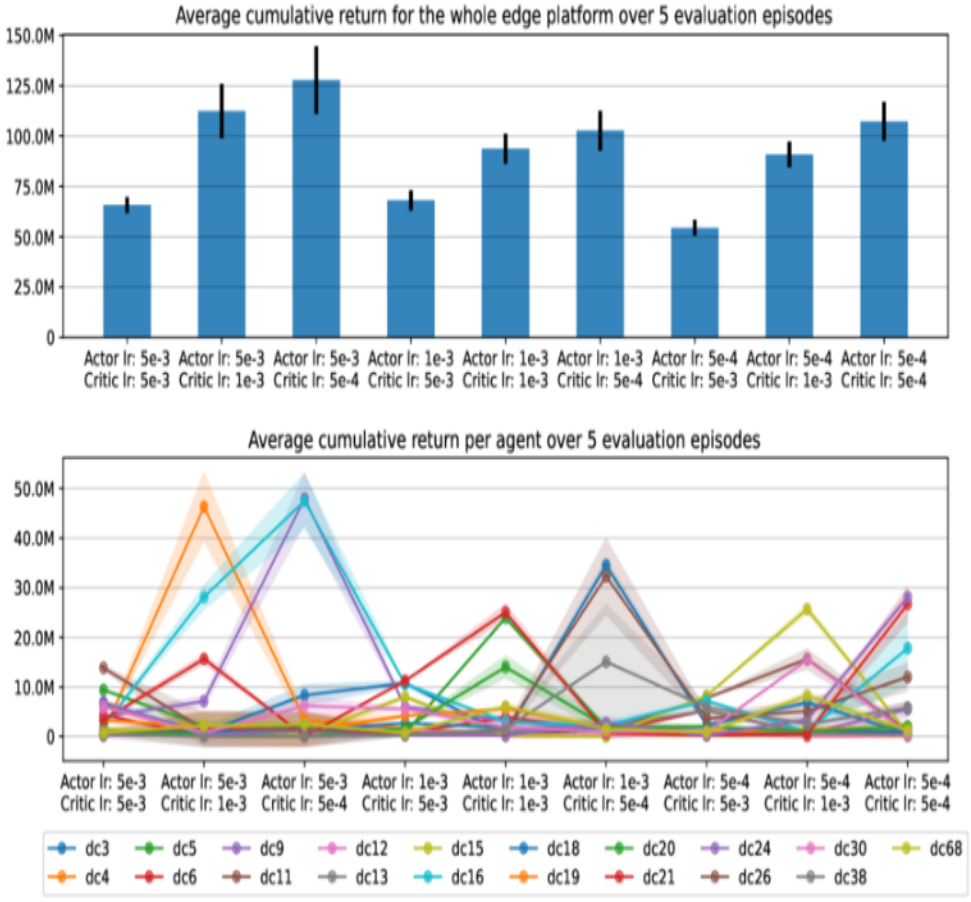

Fig : An example of a hyperparameter tuning result plot

6.1. Summary of Findings

The integration of edge and cloud computing has profound implications for the development and deployment of modern distributed applications. To achieve more effective offloading of AI tasks, we introduce a novel integration framework capable of accommodating both standard cloud-based software virtualization and processing paradigms and contemporary computing devices characteristically used in edge scenarios. This high-performance integration does not require the software at the cloud end of the system to possess detailed knowledge of the edge, and it can be used within the application system stack from a virtualization platform to a system-wide layer. By using this integration approach, we build and evaluate a feature-complete framework that can be used by distributed AI applications to scale and improve application performance while using distributed application-level edge processing resources. Through a detailed analysis, we demonstrate that the use of our framework results in more resources being available for the execution of edge applications, a significant reduction of AI task processing time due to task offloading with single-digit milliseconds' overhead when using edge-enabled storage devices, and reduces the amount of time resources need to be busy to achieve these goals. The contributions can be decomposed into several key highlights. Firstly, we demonstrated how an edge-assisted distributed application can be modeled in a way that permits effective allotment in a working system. Furthermore, detailed aspects of the application were described, such as the Snap embeddings that are used to facilitate concurrent task execution and the Atom publishing model that permits new coordination protocols to be introduced seamlessly. This intangible task model was then used to allow for a precise breakdown of application execution at each node, allowing us to identify areas that were particularly resource-constrained. Subsequently, we described an approach that permitted extensive periodical publication to be retrofitted and evaluated within each webcam node and utilized a dataset to demonstrate that the introduction of this computer will directly lead to improved perception accuracy in the end application. Lastly, we synthesized our orthogonal results to introduce an edge-resident virtualization API.

6.2. Future Trends

We forecast several trends that may be expected in the discussed convergence of edge computing and cloud computing frameworks. Firstly, the real-time processing capability of new applications is boosted by the neat interconnection with communication. Emerging tools and techniques in the field of algorithms matching heterogeneous computing servers as a distributed platform can be the frontier to drive the integration into a distributive real-time processing infrastructure system. Another future trend will be further pushed to align with user needs and industrial demands, which may result in an emerging trend of research directions. The user is still the first set to grow in the future, and the attention will not be diminished.

Integrating edge and cloud computing is an emerging research topic. In particular, edge producers can be quickly and physically close to devices collecting data and cloud data/storage resources. It is important for edge-cloud integration to provide infinite computing power and storage resources to edge layers, while green technology may significantly reduce the power consumption of future systems. However, from the impact on edge and cloud side effects, the technology trend will produce very advantageous points for companies to consider edge-cloud integration and conversion topics. As the edge grows, an overly distributed and deployed edge infrastructure has tremendous resources and high performance. The surrounding edge framework usually focuses on tree or emergency network models but is autonomous compared to other edge frameworks. Although the edge model has some disadvantageous effects, the benefits generally outweigh the detriment. Thus, by changing the research paradigm, it may be possible to properly refract the competitive edge of other edge schemes in the model. In edge and cloud convergence research, ethical issues are currently emerging as we have limited discussions available.

7. References

[1] Syed, S. Big Data Analytics In Heavy Vehicle Manufacturing: Advancing Planet 2050 Goals For A Sustainable Automotive Industry.

[2] Nampally, R. C. R. (2023). Moderlizing AI Applications In Ticketing And Reservation Systems: Revolutionizing Passenger Transport Services. In Journal for ReAttach Therapy and Developmental Diversities. Green Publication. https://doi.org/10.53555/jrtdd.v6i10s(2).3280

[3] Danda, R. R. Digital Transformation In Agriculture: The Role Of Precision Farming Technologies.

[4] Malviya, R. K., Abhireddy, N., Vankayalapti, R. K., & Sodinti, L. R. K. (2023). Quantum Cloud Computing: Transforming Cryptography, Machine Learning, and Drug Discovery.

[5] Eswar Prasad G, Hemanth Kumar G, Venkata Nagesh B, Manikanth S, Kiran P, et al. (2023) Enhancing Performance of Financial Fraud Detection Through Machine Learning Model. J Contemp Edu Theo Artificial Intel: JCETAI-101.

[6] Syed, S. (2023). Zero Carbon Manufacturing in the Automotive Industry: Integrating Predictive Analytics to Achieve Sustainable Production.

[7] Nampally, R. C. R. (2022). Neural Networks for Enhancing Rail Safety and Security: Real-Time Monitoring and Incident Prediction. In Journal of Artificial Intelligence and Big Data (Vol. 2, Issue 1, pp. 49–63). Science Publications (SCIPUB). https://doi.org/10.31586/jaibd.2022.1155

[8] Danda, R. R. Decision-Making in Medicare Prescription Drug Plans: A Generative AI Approach to Consumer Behavior Analysis.
[9] Chintale, P., Khanna, A., Desaboyina, G., & Malviya, R. K. DECISION-BASED SYSTEMS FOR ENHANCING SECURITY IN CRITICAL INFRASTRUCTURE SECTORS.
[10] Siddharth K, Gagan Kumar P, Chandrababu K, Janardhana Rao S, Sanjay Ramdas B, et al. (2023) A Comparative Analysis of Network Intrusion Detection Using Different Machine Learning Techniques. J Contemp Edu Theo Artificial Intel: JCETAI-102.
[11] Syed, S. (2023). Shaping The Future Of Large-Scale Vehicle Manufacturing: Planet 2050 Initiatives And The Role Of Predictive Analytics. Nanotechnology Perceptions, 19(3), 103-116.
[12] Nampally, R. C. R. (2022). Machine Learning Applications in Fleet Electrification: Optimizing Vehicle Maintenance and Energy Consumption. In Educational Administration: Theory and Practice. Green Publication. https://doi.org/10.53555/kuey.v28i4.8258
[13] Danda, R. R., Maguluri, K. K., Yasmeen, Z., Mandala, G., & Dileep, V. (2023). Intelligent Healthcare Systems: Harnessing Ai and Ml To Revolutionize Patient Care And Clinical Decision-Making.
[14] Rajesh Kumar Malviya , Shakir Syed , RamaChandra Rao Nampally , Valiki Dileep. (2022). Genetic Algorithm-Driven Optimization Of Neural Network Architectures For Task-Specific AI Applications. Migration Letters, 19(6), 1091–1102. Retrieved from https://migrationletters.com/index.php/ml/article/view/11417
[15] Janardhana Rao Sunkara, Sanjay Ramdas Bauskar, Chandrakanth Rao Madhavaram, Eswar Prasad Galla, Hemanth Kumar Gollangi, et al. (2023) An Evaluation of Medical Image Analysis Using Image Segmentation and Deep Learning Techniques. Journal of Artificial Intelligence & Cloud Computing. SRC/JAICC-407.DOI: doi.org/10.47363/JAICC/2023(2)388
[16] Syed, S. Advanced Manufacturing Analytics: Optimizing Engine Performance through Real-Time Data and Predictive Maintenance.
[17] RamaChandra Rao Nampally. (2022). Deep Learning-Based Predictive Models For Rail Signaling And Control Systems: Improving Operational Efficiency And Safety. Migration Letters, 19(6), 1065–1077. Retrieved from https://migrationletters.com/index.php/ml/article/view/11335
[18] Mandala, G., Danda, R. R., Nishanth, A., Yasmeen, Z., & Maguluri, K. K. AI AND ML IN HEALTHCARE: REDEFINING DIAGNOSTICS, TREATMENT, AND PERSONALIZED MEDICINE.
[19] Chintale, P., Korada, L., Ranjan, P., & Malviya, R. K. (2019). Adopting Infrastructure as Code (IaC) for Efficient Financial Cloud Management. ISSN: 2096-3246, 51(04).
[20] Gagan Kumar Patra, Chandrababu Kuraku, Siddharth Konkimalla, Venkata Nagesh Boddapati, Manikanth Sarisa, et al. (2023) Sentiment Analysis of Customer Product Review Based on Machine Learning Techniques in E-Commerce. Journal of Artificial Intelligence & Cloud Computing. SRC/JAICC-408.DOI: doi.org/10.47363/JAICC/2023(2)38
[21] Syed, S. (2022). Breaking Barriers: Leveraging Natural Language Processing In Self-Service Bi For Non-Technical Users. Available at SSRN 5032632.
[22] Nampally, R. C. R. (2021). Leveraging AI in Urban Traffic Management: Addressing Congestion and Traffic Flow with Intelligent Systems. In Journal of Artificial Intelligence and Big Data (Vol. 1, Issue 1, pp. 86–99). Science Publications (SCIPUB). https://doi.org/10.31586/jaibd.2021.1151
[23] Syed, S., & Nampally, R. C. R. (2021). Empowering Users: The Role Of AI In Enhancing Self-Service BI For Data-Driven Decision Making. In Educational Administration: Theory and Practice. Green Publication. https://doi.org/10.53555/kuey.v27i4.8105
[24] Nagesh Boddapati, V. (2023). AI-Powered Insights: Leveraging Machine Learning And Big Data For Advanced Genomic Research In Healthcare. In Educational Administration: Theory and Practice (pp. 2849–2857). Green Publication. https://doi.org/10.53555/kuey.v29i4.7531